

12

الهاكر الأخلاقي

Hacking web server

By

Dr.Mohammed Sobhy Teba
Hacking Web Server
https://www.facebook.com/tibea2004

CONTENTS

WEB SERVER CONCEPT 12.1

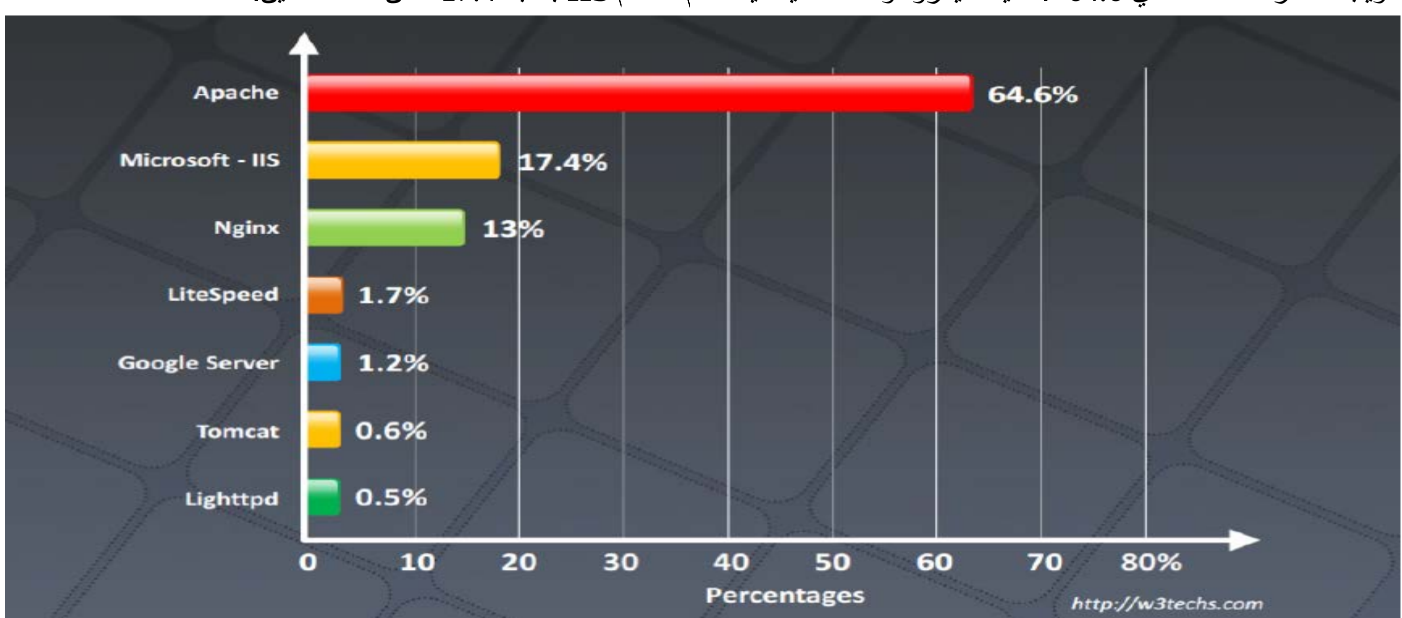
لفهم قرصنة خوادم الويب، أولا يجب أن نعرف ما هو خادم الويب، وكيف يعمل، وما هي العناصر الأخرى المرتبطة به. وتسمى كل هذه المفاهيم ببساطة خادم الويب. أولا نحن سوف نأخذ بعض الوقت في المناقشة حول مفاهيم خادم الويب.

هذا القسم يوفر لك لمحة موجزة عن خادم الويب وهندسته المعمارية. وسوف يفسر أيضا الأسباب الشائعة أو الأخطاء التي تشجع المهاجمين لاختراق خادم الويب وتصبح ناجحة في ذلك. يصف هذا القسم أيضا تأثير الهجمات على خادم الويب.

Web Server Market Shares

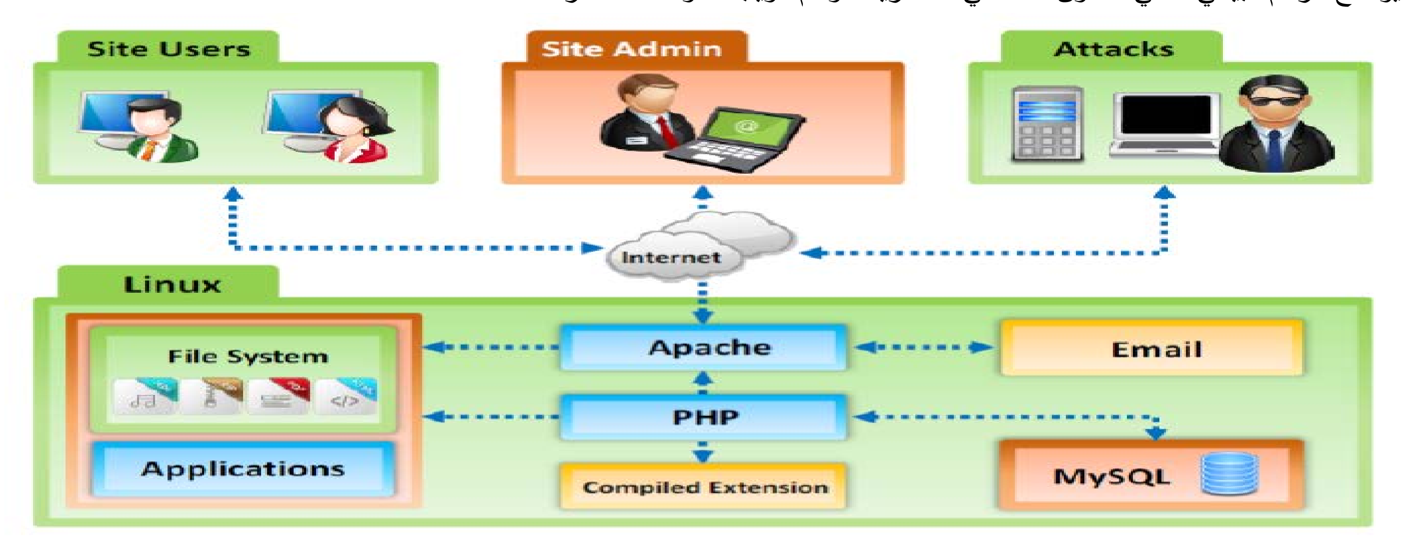
الموقع: http://w3techs.com

يبين الإحصاءات التالية النسب المئوية من المواقع باستخدام مختلف خوادم الويب. من الإحصاءات، فمن الواضح أن الأباتشي هو خادم الويب الأكثر استخداما، أي 64.6٪. تاليه مايكروسوفت – حيث يستخدم الخادم **IIS** بنسبة 17.4٪ من المستخدمين.



معمارية خوادم الويب مفتوحة المصدر "open source web server architecture"

يوضح الرسم البياني التالي المكون الأساسي لمعمارية خوادم الويب مفتوحة المصدر.



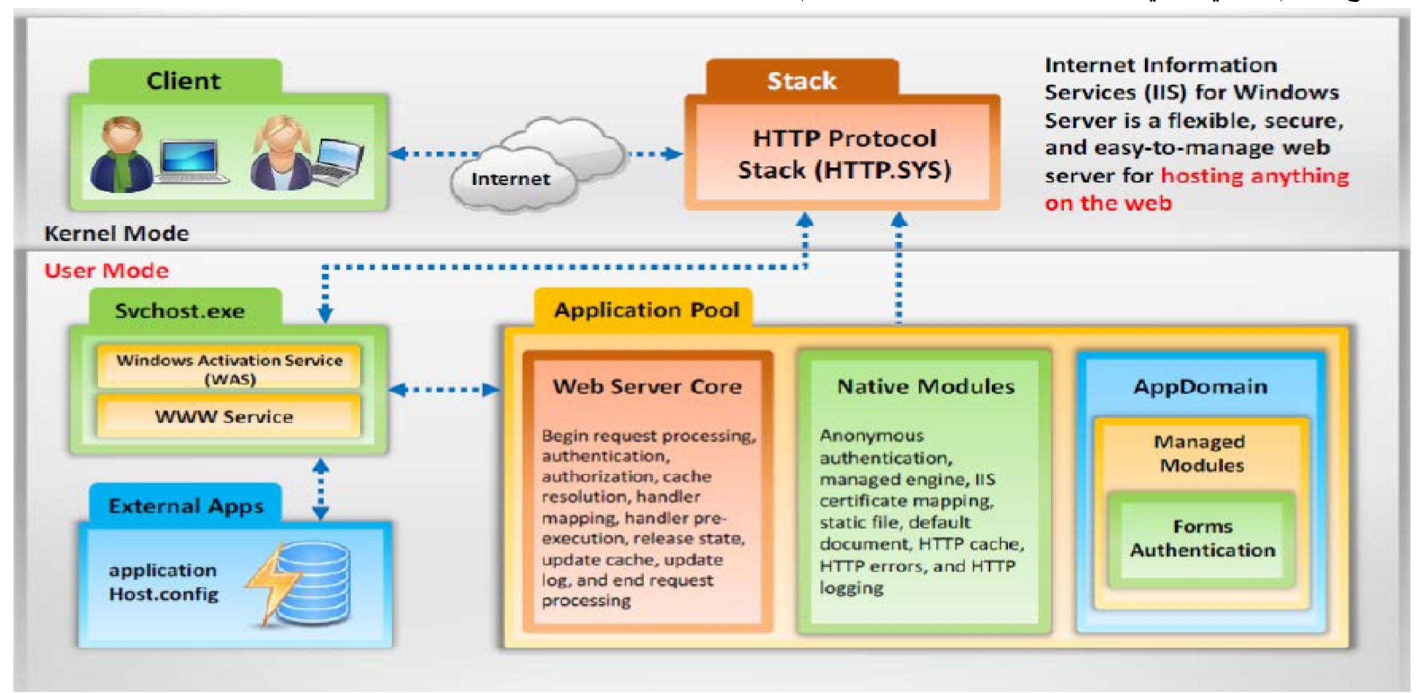
حيث ان:

Linux - the server's operating system
Apache - the web server component
MySQL - a relational database
PHP - the application layer

معمارية خوادم الويب IIS "IIS web server architecture"

IIS، المعروف أيضا باسم **Internet Information Service**، هو تطبيق خادم الويب التي طورتها شركة مايكروسوفت والتي يمكن استخدامها مع **Microsoft Windows**. وهذا هو ثاني أكبر خوادم انترنت مستخدمه بعد خادم الأباتشي **HTTP**. تحتل 17.4٪ من حصة السوق. وهو يدعم **HTTP**، **HTTPS**، **FTP**، **FTPS**، **SMTP** و**NNTP**.

يوضح الرسم البياني التالي المكونات الأساسية لمعمارية خادم الويب **IIS**:



تشويه الموقع "Website Defacement"

تشويه الموقع هو عملية تغيير محتوى الموقع على شبكة الانترنت أو صفحة على شبكة الإنترنت من قبل القراصنة. يقوم القراصنة بكسر/اختراق خوادم الويب، وتغير الموقع الذي استضافته من خلال انشاء شيء جديد.

يحدث تشويه الويب عندما يقوم المتسلل بتغير بشكل ضار المظهر المرئي للصفحة على شبكة الإنترنت عن طريق إدراج أو استبدال البيانات الاستفزازية والهجومية بشكل متكرر. الصفحات المشوه تعرض للزوار دعاية أو معلومات مضللة حتى يتم اكتشاف التغيير الغير مصرح به وتصحيحها.



لماذا خوادم الويب معرضه للاختراق

هناك مخاطر أمنية كامنة مرتبطة بخوادم الويب، والشبكات المحلية التي تستضيف المواقع على شبكة الإنترنت والمستخدمين الذين يصلون إلى هذه المواقع باستخدام المتصفحات.

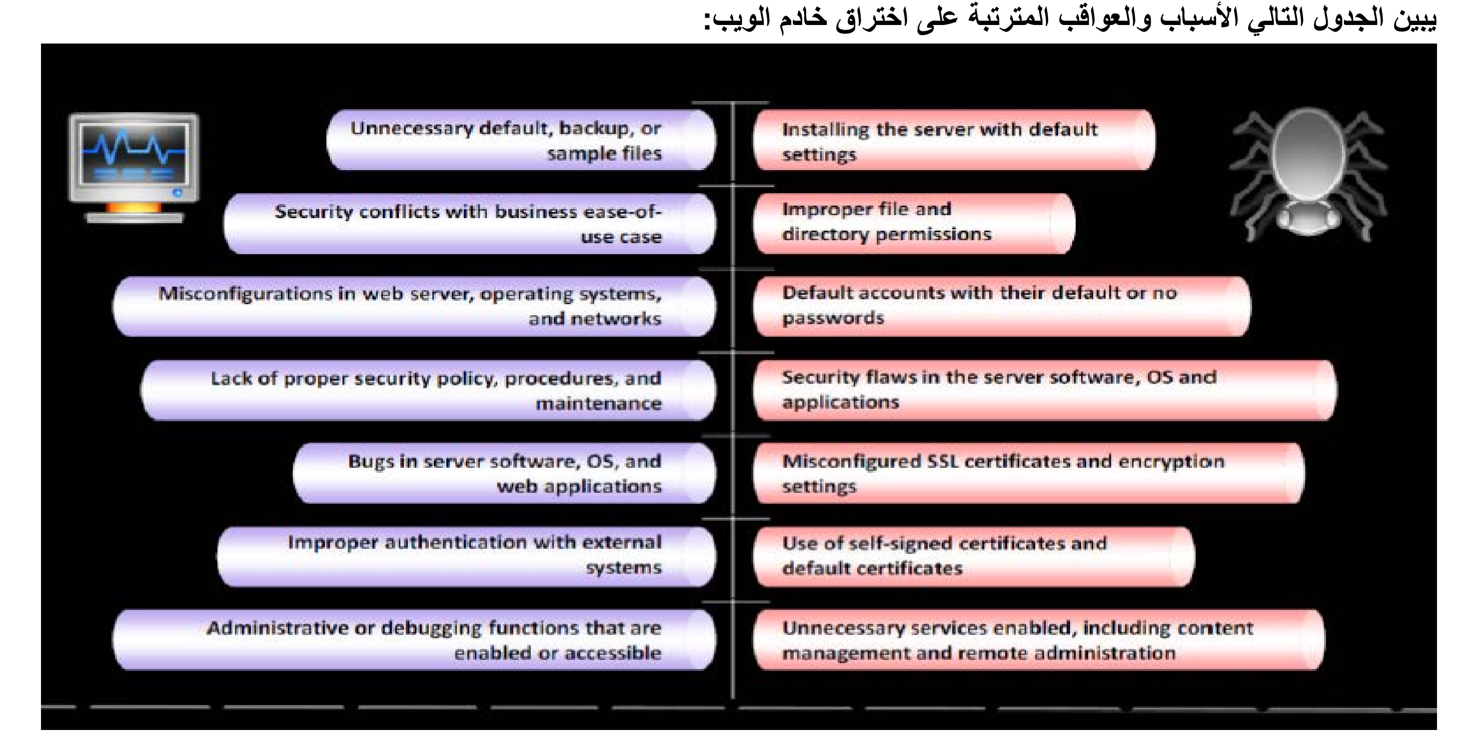
قلق مشرفي المواقع "Webmaster's Concern": من وجهة نظر مديري المواقع، فان أكثر ما يثير قلقهم الأمني هو أن خادم الويب يمكن أن يعرض الشبكة المحلية **(LAN)** أو إنترانت الشركة للتهديدات التي تشكلها شبكة الإنترنت. قد يكون هذا في شكل من اشكال الفيروسات، أحصنة طروادة، المهاجمين، أو اختراق المعلومات نفسها. غالبا ما تعتبر العيوب في البرمجيات الحالية في البرامج الكبيرة والمعقدة مصدر للثغرات الأمنية الوشيكة. ومع ذلك، خوادم الشبكة التي هي أجهزة معقدة كبيرة أيضا تأتي مع هذه المخاطر الكامنة.

وبالإضافة إلى ذلك، فإن البنية المفتوحة لخوادم الويب تتيح **arbitrary scripts** للتشغيل على جانب الملقم حين الرد على الطلبات البعيدة. أي برنامج نصي **CGI** مثبت في الموقع قد يحتوي على **bugs** والتي هي الثغرات الأمنية المحتملة.

قلق مديري الشبكة "Network Administrator's Concern": من وجهة نظر مسؤول الشبكة، الاعداد السيء لخادم الويب قد يشكل ثقب محتمل آخر في أمن الشبكة المحلية. في حين أن الهدف من الشبكة هو التحكم بالوصول إلى الشبكة، والكثير من السيطرة يمكن أن تحدث على شبكة الإنترنت وتجعله من المستحيل تقريبا استخدامه. في بيئة الإنترانت، مدير الشبكة يجب أن يكون حذرا حول اعداد خادم الويب، بحيث يتم التعرف على المستخدمين الشرعيين، والمصادقة، ومجموعات مختلفة من المستخدمين لتعيين امتيازات الوصول المتميزة.

قلق المستخدم النهائي "End User's Concern": عادة، المستخدم النهائي لا يرون/يلاحظون التهديدات الفورية، كما في **surfing the web appears** يبدو على حد سواء مأمونة ومجهولة. ومع ذلك، **active content**، مثل عناصر تحكم **ActiveX** وتطبيقات جافا، جعل من الممكن بالنسبة للتطبيقات الضارة، مثل الفيروسات، ان تغزو نظام المستخدم. الى جانب ذلك، **active content** من مستعرض موقع الويب يمكن أن يكون ممرا للبرامج الضارة لتجاوز نظام جدار الحماية والتخلل من خلال شبكة المنطقة المحلية.

يبين الجدول التالي الأسباب والعواقب المترتبة على اختراق خادم الويب:



تأثير الهجمات على خادم الويب "IMPACT OF WEB SERVER ATTACKS"

يمكن للمهاجمين التسبب في أنواع مختلفة من الأضرار التي لحقت بالمنظمة من خلال مهاجمة خادم الويب. الضرر يشمل ما يلي:

- **اختراق حساب المستخدمين "Compromise of user accounts"**: تتركز الهجمات على ملقم الويب في الغالب على اختراق حساب المستخدم. إذا كان المهاجم قادرا على اختراق حساب مستخدم، فان المهاجم يمكنه الحصول على الكثير من المعلومات المفيدة. المهاجم يمكنه استخدام حساب المستخدم لشن هجمات أخرى على خادم الويب.
- **العبث بالبيانات "Data tampering":** المهاجم يمكنه أن يغير أو يحذف البيانات. انه حتى يمكنه استبدال البيانات مع البرمجيات الخبيثة حتى أن كل من يرتبط بخادم الويب يصبح أيضا معرضا للاختراق.

- **تشويه مواقع الويب "Website defacement"**: القراصنة يمكنهم تماما تغيير شكل الموقع عن طريق استبدال البيانات الأصلية. تغيير شكل الموقع عن طريق تغيير الصور وعرض صفحات مختلفة مع رسائل خاصة بهم.
- **الهجمات الثانوية من موقع الويب "Secondary attacks from the website"**: بمجرد قيام المهاجم باختراق خادم الويب، فانه يمكن استخدام الخادم لإطلاق المزيد من الهجمات على المواقع المختلفة أو أنظمة العميل.
- **سرقة البيانات "Data theft"**: البيانات هي واحدة من الأصول الرئيسية للشركة. يمكن للمهاجمين الوصول إلى البيانات الحساسة للشركة مثل الشفرة المصدرية لبرنامج معين.
- الوصول الجذري إلى تطبيقات أخرى أو الخادم: الوصول الجذري هو أعلى امتياز لواحد يحصل على تسجيل الدخول إلى شبكة، سواء كان الخادم الخاص خادم مخصص، شبه مخصص، أو افتراضي. يمكن المهاجمين تنفيذ أي إجراء، بمجرد الحصول على الوصول الجذري إلى المصدر.

12.2 الهجمات على مواقع الويب "WEBSERVER ATTACKS"

بالنظر إلى أنك أصبحت على دراية بمفاهيم خوادم شبكة الإنترنت، فنحن نتحرك قدما إلى الهجمات المحتملة على خادم الويب. حيث يتم تنفيذ كل عمل على الانترنت مع مساعدة من خادم الويب. وبالتالي، فإنه يعتبر مصدر حرج للمنظمة. وهذا هو نفس السبب الذي يجعل المهاجمين يستهدفون خادم الويب. هناك العديد من تقنية الهجوم المستخدمة من قبل المهاجم لاختراق خادم الويب. الآن سوف نناقش حول تقنيات الهجوم تلك.

التكوين الخاطئ لخادم الويب "WEB SERVER MISCONFIGURATION"

خوادم الويب لديها العديد من نقاط الضعف المتعلقة بالتكوين، التطبيقات، الملفات، الاسكريبات، أو صفحات الويب. بمجرد العثور على هذه الثغرات من قبل المهاجم، مثل الوصول عن بعد للتطبيق، فتصبح هذه المداخل المدخل للمهاجمين للدخول الى شبكة الشركة. ويمكن لهذة الثغرات من الخادم مساعدة المهاجمين لتجاوز مصادقة المستخدم. التكوين الخاطئ للخادم يشير إلى تكوين نقاط الضعف في البنية التحتية على شبكة الإنترنت التي يمكن استغلالها لشن هجمات مختلفة على خوادم الويب مثل **directory traversal**، **server intrusion**، وسرقة البيانات. بمجرد اكتشاف هذه المشاكل يمكن استغلالها بسهولة ويؤدي إلى الاختراق الكلي للموقع على شبكة الانترنت.

- وظائف الوصول البعد للمسئولين "**Remote administration functions**" يكون مصدرا لكسر الخادم من قبل المهاجمين.
- بعض الخدمات الغير الضرورية تعتبر هي أيضا نقطة ضعف بالنسبة للقرصنة.
- التكوين الخاطئ/الافتراضي لشهادات **SSL**.
- رسائل التصحيح/الخطأ.
- اسم المستخدم/كلمات السر لل **anonymous** والافتراضي.
- عينات التكوين وملفات الاسكريبت.



Web Server Misconfiguration Example

بالنظر الى ملف الاعداد **httpd.conf** التالي الخاص بخادم الويب أباتشي.

```
<Location /server-status>
SetHandler server-status
</Location>
```

هذا التكوين يسمح لأي شخص بعرض صفحة حالة الملقم الذي يحتوي على معلومات مفصلة حول الاستخدام الحالي لخادم الويب، بما في ذلك معلومات عن المضيفين والطلبات الحالية قيد المعالجة.

بالنظر الى مثال اخر، ملف الاعداد **php.ini** كالاتي:

```
display_error = On
log_errors = On
error_log = syslog
ignore_repeated_errors = Off
```

هذا التكوين يعطي رسائل الخطأ **verbose error messages**.

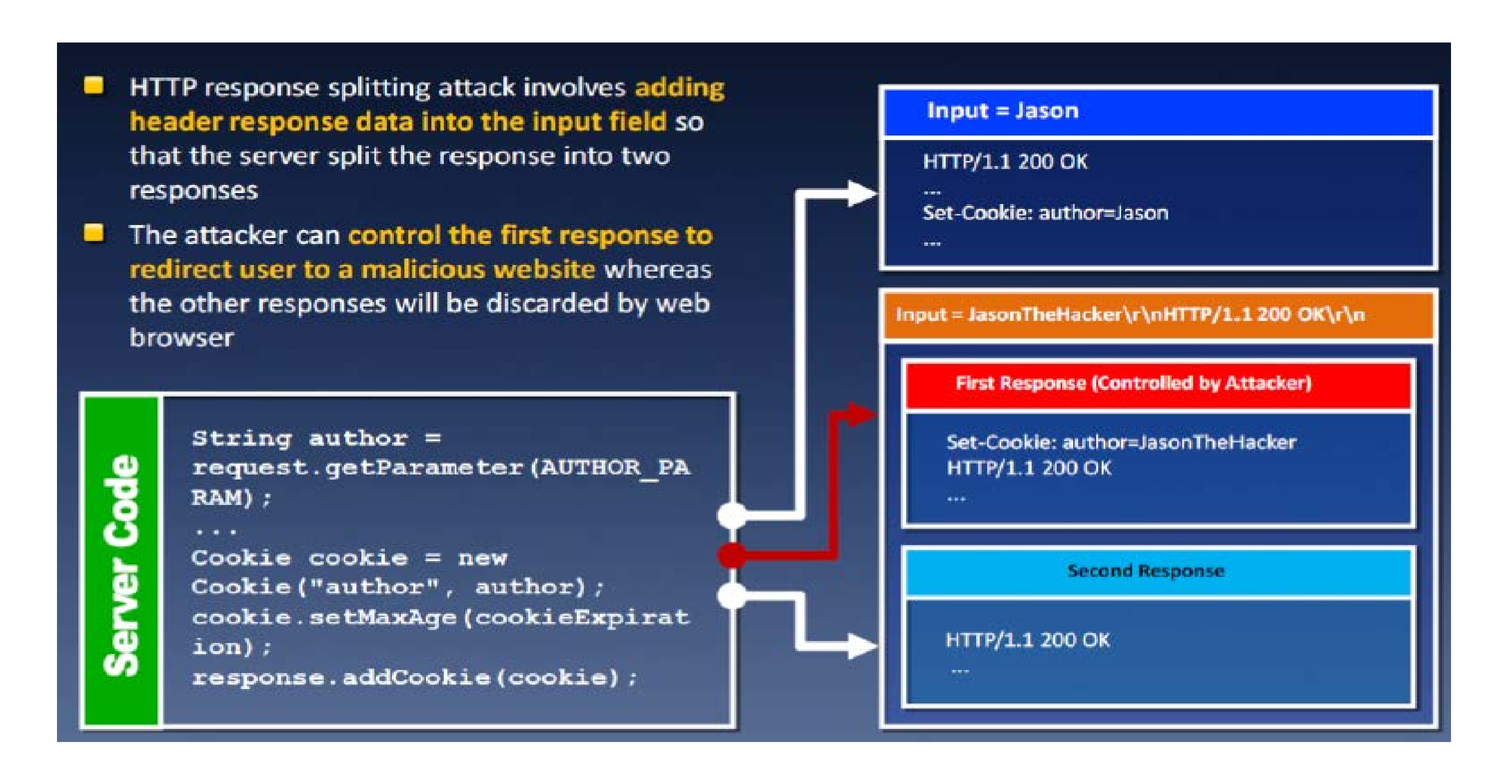
هجمات اجتياز الدليل "Directory Traversal Attacks"

تم تصميم خوادم الشبكة بطريقة ما لتقيد وصول الجمهور إلى حد ما. هجمات اجتياز الدليل "**Directory Traversal Attacks**" هو استغلال **HTTP** والتي تجعل المهاجمين قادرون على الوصول إلى المجلدات المقيدة وتنفيذ الأوامر خارج المجلد الجذري لخادم الويب عن طريق التلاعب في **URL**. يمكن المهاجمين استخدام أسلوب التجربة والخطأ "**trial-and-error**" للتنقل خارج الدليل الجذري والوصول إلى المعلومات الحساسة في النظام.



HTTP RESPONSE SPLITTING ATTACK

هجوم **HTTP response attack** هو هجوم على شبكة الإنترنت حيث يتم خداع الخادم عن طريق حقن خطوط جديدة إلى رؤوس الاستجابة جنبا إلى جنب مع تعليمات برمجية عشوائية. **Cross-Site Scripting (XSS)**، **Cross Site Request Forgery (CSRF)**، و **SQL Injection** هي بعض من الأمثلة لهذا النوع من الهجمات. المهاجم يغير الطلب الواحد ليظهر وتتم معالجته من قبل خادم الويب عن انه اثنين من الطلبات. خادم الويب بدوره يستجيب لكل طلب. ويتم إنجاز هذا عن طريق إضافة بيانات راس الاستجابة " **response header**" في حقل الإدخال. المهاجم يمرر البيانات الخبيثة الى نقاط ضعف التطبيقات، والتطبيق الذي يتضمن البيانات في رأس استجابة **HTTP**. يمكن للمهاجم التحكم في أول رد لإعادة توجيه المستخدم إلى موقع خبيثة، في حين انه سيتم تجاهل الاستجابات الأخرى الى متصفح الويب.

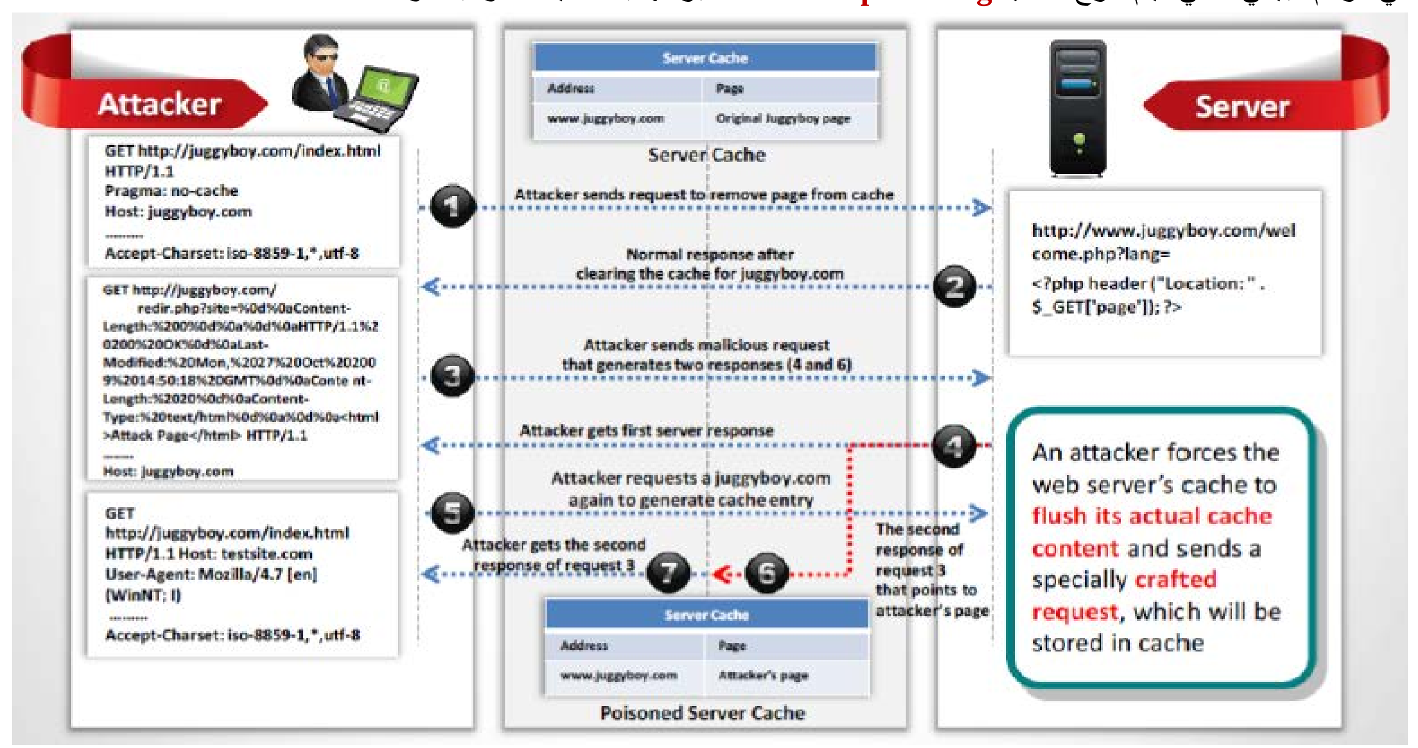


WEB CACHE POISONING ATTACK هجوم

Web cache poisoning هو هجوم الذي نفذ على النقيض من موثوقية **intermediate web cache source**، والتي يتم فيها التبادل للمحتوى الصادق لـ **URL** عشوائي "**honest content cached for a random URL**" مع المحتوى المصاب. مستخدمي الويب **web cache source** لا يدرون انهم يستخدمون المحتوى **poisoned content** بدلا من المحتوى الحقيقي والمضمون عندما يطلب **URL** المطلوبة من خلال **web cache**.

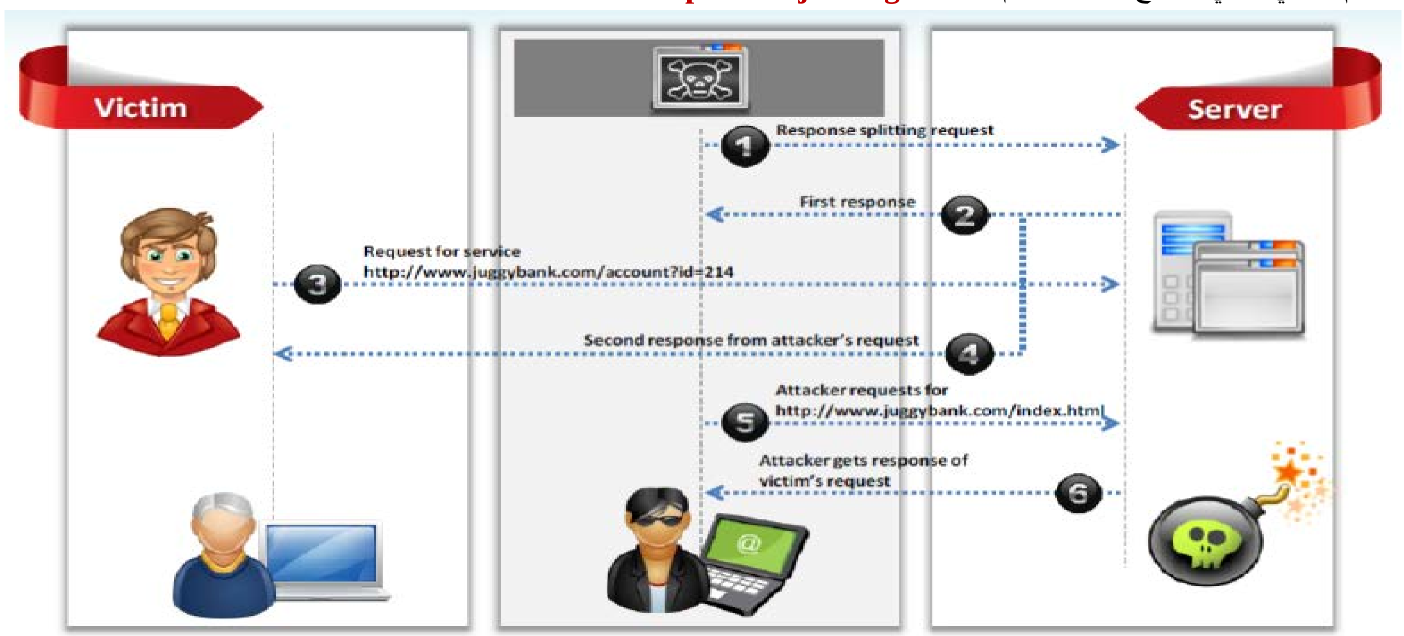
يجبر المهاجم **web server's cache** على طرد محتوى **cache** الفعلي الخاص به وإرسال طلب وضع خصيصا لتخزينه في **cache**.

في الرسم البياني التالي، يتم شرح العملية **web cache poisoning** برمتها بالتفصيل خطوة بخطوة.



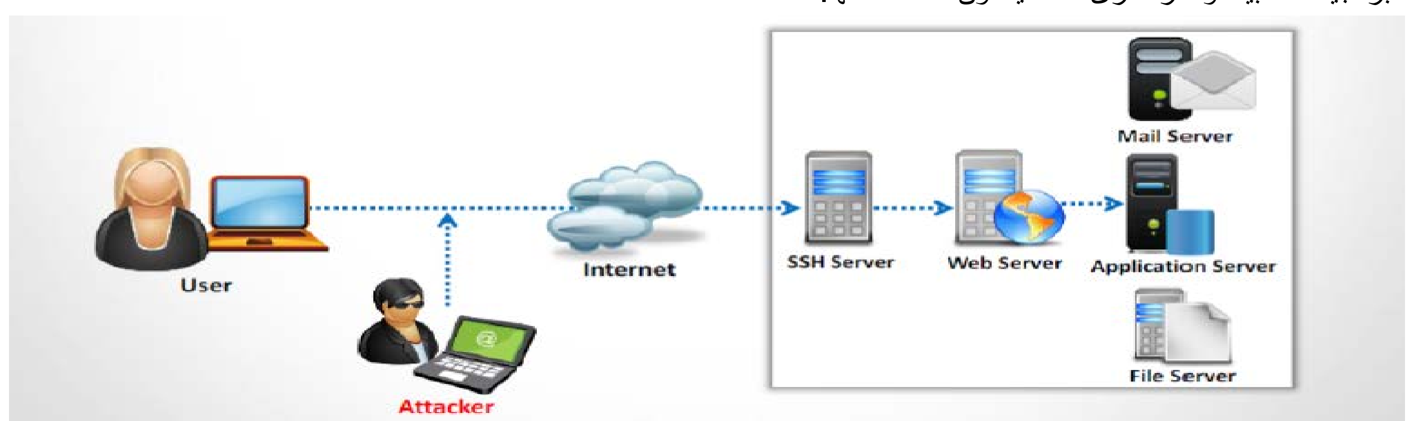
HTTP RESPONSE HIJACKING

يتم إنجاز **HTTP response hijacking** مع **response splitting request**. في هذا الهجوم، في البداية يقوم المهاجم بإرسال طلب **response splitting request** الى خادم الويب. يقوم الخادم بتقسم الاستجابة إلى قسمين ويرسل أول رد إلى المهاجم والاستجابة الثانية للضحية. بمجرد تلقي الرد من خادم الويب، فان الضحية يستجيب للخدمة من خلال منح وثائق التفويض. في الوقت نفسه، يطلب المهاجم صفحة الفهرس "***index page***". ثم بعد ذلك يقوم خادم الويب بإرسال استجابة طلب الضحية إلى المهاجم ويبقى ضحية جاهل لما يحدث. الرسم البياني التالي يوضح إجراء هجوم **HTTP response hijacking attack** خطوة بخطوة:



هجوم SSH Brute Force Attack

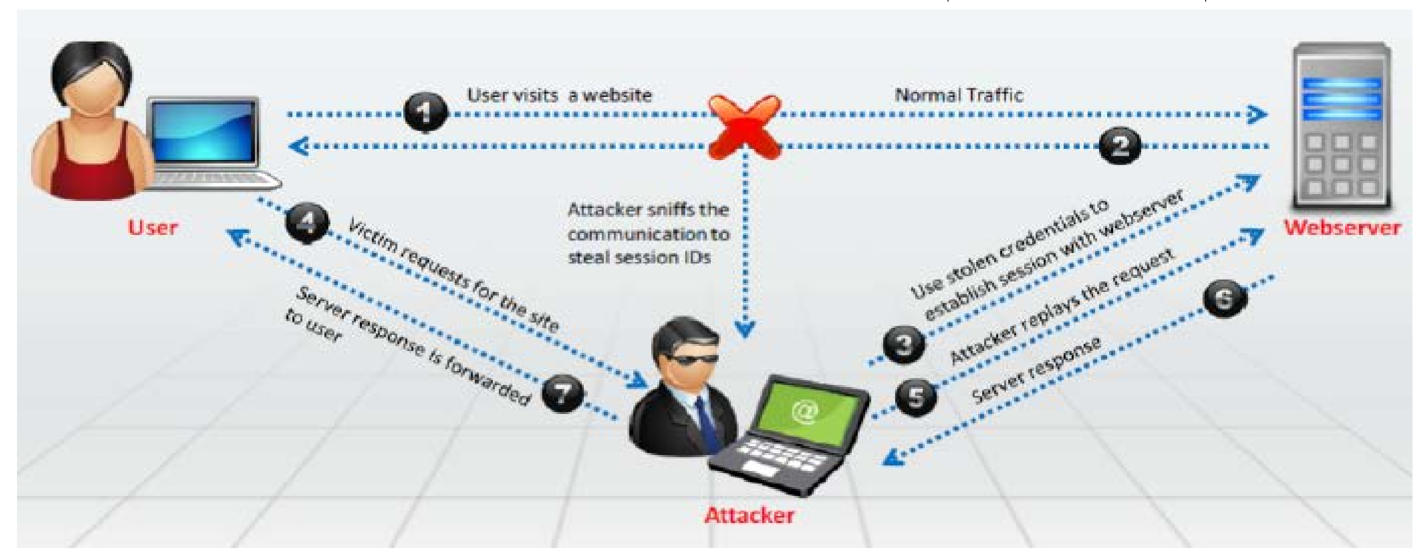
يستخدم البروتوكول **SSH** لإنشاء **SSH tunnel** بين اثنين من المضيفين من أجل نقل البيانات الغير مشفرة عبر شبكة آمنة. من أجل إجراء هجوم على **SSH**، أولا المهاجم يقو م بفحص خادم **SSH** كامل لتحديد نقاط الضعف المحتملة. مع مساعدة من هجوم القوة الغاشمة، المهاجم يكسب بيانات اعتماد تسجيل الدخول. وبمجرد كسب المهاجم اعتماد تسجيل الدخول من **SSH**، فأنه يستخدم نفس الأنفاق **SSH** لنقل البرمجيات الخبيثة ومآثر أخرى لضحايا دون الكشف عنها.



MAN-IN-THE-MIDDLE ATTACK

هجوم رجل-في-الوسط هي طريقة التي يكون فيها اعتراض الدخيل أو تعديل الرسالة التي يتم تبادلها بين المستخدم وخادم الويب من خلال التنصت أو التداخل في الاتصال. وهذا يسمح للمهاجمين بسرقة المعلومات الحساسة من المستخدم مثل التفاصيل المصرفية عبر الإنترنت،

وأسماء المستخدمين وكلمات السر، الخ التي يتم نقلها عبر الإنترنت إلى خادم الويب. المهاجم يخدع الضحية للاتصال بخادم الويب من خلال التظاهر بانه بروكسي. إذا اعتقد الضحية ووافق على طلب المهاجم، فان كل الاتصالات بين المستخدم وخادم الويب يمر عبر المهاجم. وهكذا، يمكن للمهاجم سرقة معلومات المستخدم الحساسة.



كسر كلمة السر لخادم الويب "WEB SERVER PASSWORD CRACKING"

تبدأ معظم القرصنة مع كسر كلمة السر فقط. بمجرد كسر كلمة السر، يمكن للهاكر الدخول في شبكة كشخص مفوض. معظم الكلمات الأكثر شيوعا وجدت كأسماء لكلمة السر **password**، **root**، **administrator**، **admin**، **demo**، **test**، **guest**، **QWERTY**، أسماء الحيوانات الأليفة، الخ. المهاجمون يستخدمون أساليب مختلفة مثل الهندسة الاجتماعية، والخداع، التصيد، وذلك باستخدام حصان طروادة أو فيروس، التنصت، راصد لوحة مفاتيح، هجوم القوة الغاشمة، هجوم القاموس، وما إلى ذلك لكسر كلمات السر.

المهاجمين يستهدفون أساسا:

- Web form authentication cracking
- SSH tunnels
- FTP servers
- SMTP servers
- Web shares

تقنيات كسر كلمات المرور لخوادم الويب "Web Server Password Cracking Techniques"

قد يتم كسر كلمات السر يدويا أو مع الأدوات الآلية مثل **Cain & Abel**، **Brutus**، **THC Hydra**، الخ. المهاجمون يتبع تقنيات مختلفة لكسر كلمة السر:

التخمين "guessing": طريقة التكسير الشائعة والمستخدمة من قبل المهاجمين هو تخمين كلمات السر إما من قبل البشر أو عن طريق الأدوات الآلية المقدمة مع القواميس. معظم الناس يميلون إلى استخدام أسماء أحبائهم، أسماء حيواناتهم الأليفة، أرقام لوحات السيارات، وتواريخ الميلاد، أو غيرها من الكلمات الضعيفة مثل **"QWERTY"**، **"Password"**،**"admin"**، وغيرها حتى يتمكنوا من تذكرها بسهولة. الشيء نفسه يسمح للمهاجم بكسر كلمات السر عن طريق التخمين.

هجوم القاموس "dictionary attack": هجوم القاموس هو الأسلوب المستخدم لتوليف العديد من الكلمات المختلفة، ولكن هذا قد لا يكون من الممكن أيضا أن تكون فعالة إذا كانت كلمة المرور تتكون من الحروف والرموز الخاصة، ولكن بالمقارنة مع هجوم القوة الغاشمة فهذا أقل استهلاكا للوقت.

هجوم القوة الغاشمة "Brute Force Attack": في أسلوب القوة الغاشمة، يتم اختبار كل الحروف الممكنة، على سبيل المثال، الأحرف الكبيرة من "A إلى Z" أو أرقام من "0-9" أو الصغيرة من "a الى z". ولكن هذا النوع من الطريقة مفيدة لتحديد كلمة واحدة أو كلمتين مرور. في حين إذا كانت كلمة مرور تتكون من الأحرف الكبيرة والصغيرة والحروف الخاصة، فإنه قد يستغرق شهورا أو سنوات لكسر كلمة السر، وهو أمر مستحيل عمليا.

الهجوم الهجين "hybrid attack": الهجوم الهجين هو أكثر قوة كما أنه يستخدم كل من هجوم القاموس وهجوم القوة الغاشمة. وهو يتألف أيضا من الرموز والأرقام. يصبح كسر كلمة السر أسهل مع هذا الأسلوب.

الهجوم على تطبيقات الويب "WEB APPLICATION ATTACK"

نقاط الضعف في تطبيقات الويب التي تعمل على خادم الويب توفر مسار واسع من الهجوم لاختراق خادم الويب.

- **اجتياز الدليل "Directory Traversal"**

اجتياز الدليل هو استغلال **HTTP** والتي من خلالها يكون المهاجمين قادرون على الوصول إلى المجلدات المقيدة وتنفيذ الأوامر خارج المجلد الجذري لخادم الويب عن طريق التلاعب في **URL**.

- **Parameter/Form Tampering**

يهدف هذا النوع من الهجوم على التعامل مع المعلمات المتبادلة بين العميل والخادم من أجل تعديل بيانات التطبيق، مثل أوراق اعتماد المستخدم والأذونات وسعر وكمية المنتجات، الخ.

- **Cookie Tampering**

هو أسلوب من **poisoning** أو **tampering** بملفات كوكيز الخاصة بالعميل. معظم مراحل هذا الهجوم تتم عند إرسال ملفات كوكيز من جانب العميل إلى الملقم. ملفات كوكيز الدائمة وغير الدائمة يمكن تعديلها باستخدام أدوات مختلفة.

- **Command Injection Attacks**

هو هجوم الذي يقوم فيه القراصنة بتغير محتوى صفحة الويب باستخدام الكود وعن طريق تحديد حقول النموذج التي تفتقر القيود الصالحة.

- **Buffer Overflow Attacks**

تم تصميم معظم تطبيقات الويب للحفاظ على كميات من البيانات. إذا تم تجاوز هذا، يحدث عطل للتطبيق أو قد تظهر بعض سلوك الضعف للآخرين. يستخدم المهاجم هذه الميزة وفيضانات التطبيقات من خلال الكثير من البيانات، وهذا بدوره يؤدي إلى هجوم **buffer overflow**.

- **Cross-Site Scripting (XSS) Attacks**

هي طريقة المهاجمين لحقن **HTML tags** أو سكريبت في موقع على شبكة الانترنت الهدف.

- **هجوم الحرمان من الخدمة (DoS)**

هذا الهجوم هو شكل من الأشكال الذي يهدف فيه المهاجمين إلى إنهاء عمليات موقع على شبكة الانترنت أو ملقم وجعلها غير متاحة لوصول المستخدمين الصالحين.

- **Unvalidated Input and File injection Attacks**

هذه الهجمات تشير إلى الهجمات التي تقوم بتحميل **unvalidated input** أو عن طريق حقن الملفات إلى تطبيق ويب.

- **Cross-Site Request Forgery (CSRF) Attack**

في هذا الهجوم تقوم صفحة ويب خبيثة تطلب من متصفح الويب المستخدم بإرسال الطلبات إلى الموقع على شبكة الانترنت الخبيثة حيث يتم تنفيذ مختلف الإجراءات الضعيفة، والتي ليس الغرض منها المستخدم. هذا النوع من الهجوم هو خطير في حالة المواقع المالية.

- **SQL Injection Attacks**

هو تقنية لحقن اكواد تستخدم ثغرة أمنية موجودة في قاعدة بيانات للهجمات. المهاجم يحقن الشيفرات الخبيثة في السلاسل التي في وقت لاحق تنتقل إلى **SQL Server** للحصول على التنفيذ.

- **Session Hijacking**

هو هجوم حيث يستغل المهاجم، يسرق، يتنبأ، ويتفاوض مع آلية الرقابة لجلسة الويب الصالحة الحقيقية للوصول إلى أجزاء موثقة من تطبيق الويب.

12.3 منهجية الهجوم "ATTACK METHODOLOGY"

حتى الآن ناقشنا مفاهيم خادم الويب ومختلف التقنيات التي يستخدمها المهاجم لاختراق خادم الويب. المهاجمون عادة يقوم المهاجمين باختراق خادم الويب باتباع أسلوب إجرائي. الآن سوف نناقش منهجية الهجوم التي استخدمها المهاجمون في اختراق خوادم الشبكة. يقدم هذا القسم نظرة ثاقبة على المنهجية والأدوات التي تساعد في مراحل مختلفة من الهجوم.

منهجية الهجوم على خادم الويب على شبكة الإنترنت

قرصنة خادم الويب يتم إنجازها في مختلف المراحل. في كل مرحلة يحاول المهاجم جمع المزيد من المعلومات حول الثغرات ويحاول الوصول الغير مصرح به إلى خادم الويب. مراحل الهجوم على خادم الويب تشمل:

- **جمع المعلومات "information gathering"**

يحاول كل من المهاجمين جمع أكبر قدر ممكن من المعلومات عن خادم الويب الهدف. بمجرد ان يتم جمع المعلومات، فانه يحلل المعلومات التي تم جمعها من أجل العثور على ثغرات أمنية في الآلية الحالية لخادم الويب.

- **Web Server Footprinting**

الغرض من **Footprinting** هو جمع المزيد من المعلومات حول الجوانب الأمنية لخادم الويب مع مساعدة من أدوات أو تقنيات البصمة. والغرض الرئيسي هو معرفة قدرات الوصول عن بعد ومنافذها والخدمات، وجوانب أمنها.

- **Mirroring Website**

هي وسيلة لنسخ الموقع ومحتواه على خادم آخر للتصفح دون اتصال.

- **فحص نقاط الضعف "Vulnerability Scanning"**

فحص نقاط الضعف هي وسيلة لإيجاد مختلف نقاط الضعف والاعداد الخاطئ في خادم ويب. ويتم فحص نقاط الضعف بمساعدة مختلف الأدوات الآلية المعروفة باسم **vulnerable scanners**.

- **اختطاف الجلسة "session hijacking"**

اختطاف الجلسة ممكن بمجرد ان يتم التعرف على الجلسة الحالية للعميل. يتم أخذ السيطرة الكاملة على جلسة عمل المستخدم من قبل المهاجم عن طريق اختطاف الجلسة.

- قرصنة كلمات السر لخادم ويب

استخدام المهاجمين مختلف الأساليب لكسر كلمة المرور مثل هجمات القوة الغاشمة، والهجمات الهجينة، هجمات القاموس، وغيرها، لكسر كلمات السر لخادم الويب.

Web Server Attack Methodology: Information Gathering

كل المهاجم قبل البدء في عملية القرصنة يقوم أولا بجمع كافة المعلومات المطلوبة مثل الإصدارات والتقنيات المستخدمة من قبل خادم الويب، الخ. المهاجمون يبحثون في الإنترنت في مجموعات الأخبار، لوحات الإعلانات، وما إلى ذلك للحصول على معلومات عن الشركة. المهاجمين يقضون معظم الوقت في مرحلة جمع المعلومات فقط. هذا هو السبب في ان جمع المعلومات على حد سواء فنا فضلا عن العلوم. هناك العديد من الأدوات التي يمكن استخدامها لجمع المعلومات أو الحصول على تفاصيل مثل اسم الدومين، عنوان **IP**، أو **autonomous system number**. وتشمل الأدوات:

- Whois
- Traceroute
- Active Whois
- Nmap
- Angry IP Scanner
- Netcat

- ***Whois***

المصدر: https://www.whois.net

Whois يسمح لك لأداء بحث **Whois** عن الدومين وبحث **Whois** عن **IP** والبحث في قاعدة بيانات **Whois** عن المعلومات ذات الصلة على النطاق المسجل ومدى توافرها. وهذا يمكن أن يساعد على توفير نظرة ثاقبة عن تاريخ الدومين ومعلومات إضافية. ويمكن استخدامه لإجراء بحث لمعرفة من الذي يملك اسم الدومين، كم عدد الصفحات المتوفرة على موقع جوجل، أو حتى البحث في قوائم عناوين **Whois** لمعرفة صاحب الموقع على شبكة الانترنت.



Web Server Attack Methodology: Web server Footprinting

الغرض من **Footprinting** هو جمع تفاصيل عن الحساب، نظام التشغيل وإصدارات البرامج الأخرى، وأسماء الخادم، وتفاصيل مخطط قاعدة البيانات وأكبر قدر من المعلومات الممكنة حول الجوانب الأمنية لخادم الويب أو الشبكة المستهدفة. الغرض الرئيسي هو لمعرفة قدرات الوصول عن بعد، والمنافذ والخدمات المتاحة، والآليات الأمنية المنفذة. **Telnet** لخادم الويب من اجل **Footprinting** لخادم الويب وجمع معلومات مثل اسم الخادم، نوع الخادم وأنظمة التشغيل والتطبيقات التي تعمل، وما إلى ذلك. أمثلة من الأدوات المستخدمة لأداء **Footprinting** تشمل **ID Serve**، **httprecon**، **Netcraft**، الخ.

- ***Netcraft***

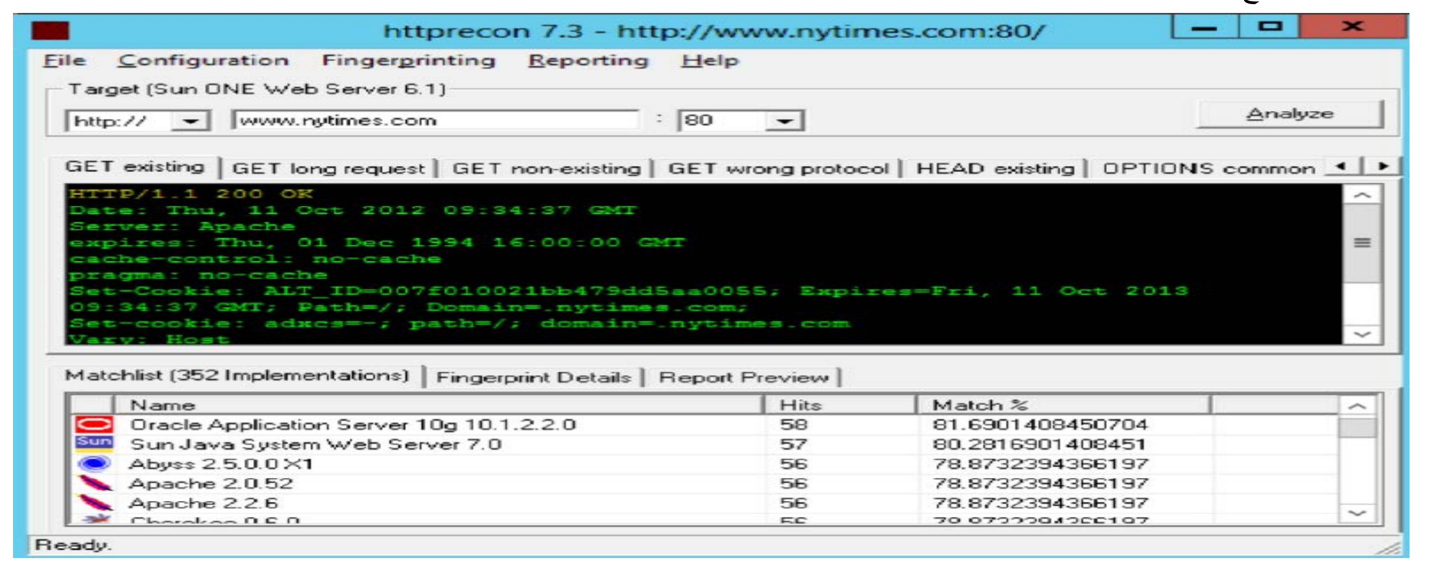
المصدر: http://toolbar.netcraft.com

نيتكرافت هي أداة تستخدم لتحديد أنظمة التشغيل قيد الاستخدام من قبل المنظمة المستهدفة. وقد سبق بيان ذلك بالتفصيل في الجزء الخاص بـ **Footprinting** و**Reconnaissance**. بالإضافة إلى أداة نيتكرافت، هناك اثنين من أكثر الأدوات التي تسمح لك لأداء **Footprinting** لخادم الويب. هم **Httprecon** و**ID service**.

- ***Httprecon***

المصدر: http://www.computec.ch

Httprecon هو أداة لأداء **Footprinting** لخادم الويب متقدمة. يقوم المشروع **httprecon** بعض الأبحاث في مجال **Footprinting** لخادم الويب، والمعروف أيضا بـ **http fingerprinting**. الهدف هو تحديد دقيق للغاية لـ **httpd** المطبقة. هذا البرنامج تحسن بسهولة وكفاءة هذا النوع من **enumeration**.



- ***ID Serve***

المصدر: https://www.grc.com/intro.htm

ID serve هي أداة بسيطة لتحديد ملقم إنترنت. **ID serve** يمكن تحديد دائما الصانع، الطراز، ونسخة برنامج الخادم لأي موقع على شبكة الانترنت. وعادة ما يتم إرسال هذه المعلومات في ديباجة الردود على الاستفسارات على شبكة الإنترنت، ولكن لا تبين للمستخدم. **ID serve** يمكنها أيضا التواصل مع خوادم الغير ويب لتلقي وتقديم تقارير رسالة تحية الخادم. هذا يكشف بشكل عام الملقم والطراز، نسخة، وغيرها من معلومات قد تكون مفيدة. ببساطة عن طريق إدخال أي عنوان **IP**، **ID serve** سيحاولون تحديد اسم الدومين المرتبط به.



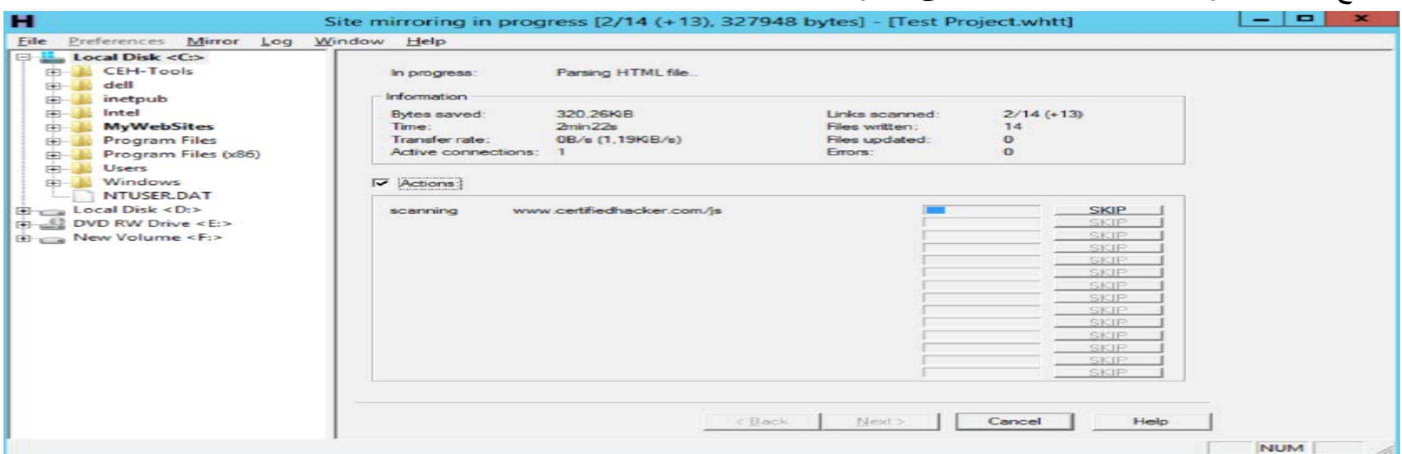
Web Server Attack Methodology: Mirroring a Website

Website mirroring هي وسيلة لنسخ الموقع ومحتواه الى خادم آخر. بواسطة ***Website mirroring*** لموقع على شبكة الانترنت، يتم إنشاء ملف تعريف كامل لبنية الموقع، بنية الملف، والارتباطات الخارجية، الخ. بمجرد ان يتم إنشاء ***Website mirroring***، والبحث عن التعليقات وغيرها من البنود الموجودة في التعليمات البرمجية لمصدر **HTML** لجعل أنشطة **Footprinting** أكثر كفاءة. الأدوات المختلفة المستخدمة في **web server mirroring** تشمل **HTTrack**، **Webripper 2.0**، **WinWSD**، **Webcopier**، و**Blackwidow**.

- ***HTTrack***

المصدر: http://www.httrack.com

HTTrack هي أداة لتصفح الموقع **offline**. انها تسمح لك بتحميل موقع الشبكة العالمية من الإنترنت إلى الدليل المحلي، وبناء متكرر لكل المجلدات، والحصول على **HTML** والصور وغيرها من الملفات من الخادم إلى جهاز الكمبيوتر الخاص بك. **HTTrack** يرتب هيكل الروابط النسبية للموقع الأصلي. ببساطة فتح صفحة من "**mirrored**" لموقع على الانترنت في المتصفح الخاص بك، ويمكنك تصفح الموقع من وصلة ربط، كما لو كنت تشاهده على الانترنت.



Web Server Attack Methodology: Vulnerability Scanning

فحص نقاط الضعف هي طريقة لتحديد نقاط الضعف المختلفة والخاطئة من خادم الويب أو الشبكة المستهدفة. ويتم فحص نقاط الضعف بمساعدة مختلف الأدوات الآلية المعروفة باسم **vulnerable scanners**.

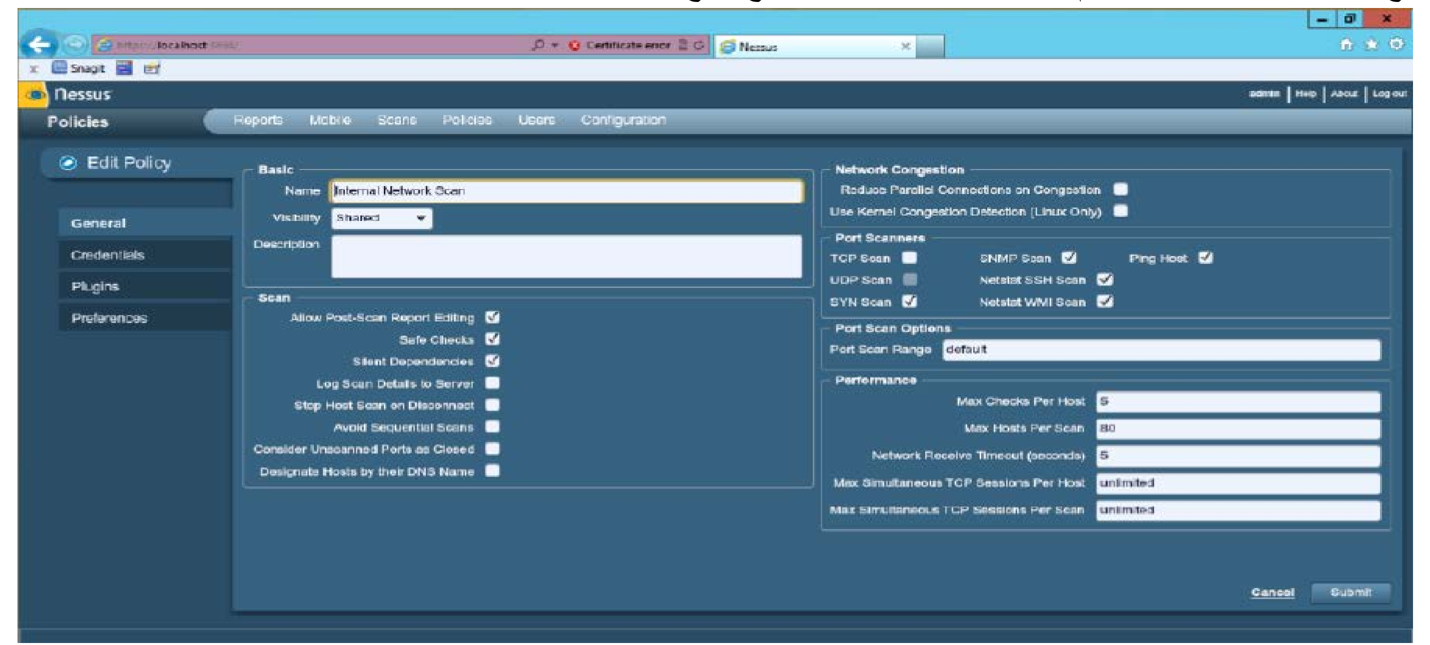
فحص نقاط الضعف يسمح لتحديد نقاط الضعف الموجودة في خادم الويب والتكوين الخاص به. وهكذا، فإنه يساعد على تحديد ما إذا كان خادم الويب هو **exploitable** أم لا. واعتمدت تقنيات التنصت على حركة مرور الشبكة لمعرفة الأنظمة النشطة، خدمات الشبكة، والتطبيقات، ونقاط الضعف الحالية.

أيضا، المهاجمين يقومون باختبار البنية التحتية لخادم الويب لأي تكوين خاطئ، والمحتوى الذي عفا عليه الزمن، ونقاط الضعف المعروفة. تستخدم أدوات مختلفة لفحص نقاط الضعف مثل **HP WebInspect**، **Nessus**، **Paros proxy**، وما إلى ذلك. لتجد المضيفين، والخدمات، ونقاط الضعف.

- ***Nessus***

المصدر: http://www.tenable.com/products/nessus

Nessus هي أدوات الفحص الأمنية التي تفحص النظام عن بعد وتعطى تقارير إذا اكتشف أي من نقاط الضعف قبل قيام المهاجم بالهجمات واختراقه له. وتشمل خمسة مزايا هي **high-speed discovery**، **configuration auditing**، **asset profiling**، **sensitive data discovery**، **patch management integration**، وتحليل هشاشة الوضع الأمني الخاص بك "***vulnerability analysis***" مع ميزات تعزز سهولة الاستخدام والفعالية والكفاءة، والتواصل مع جميع أجزاء المؤسسة.



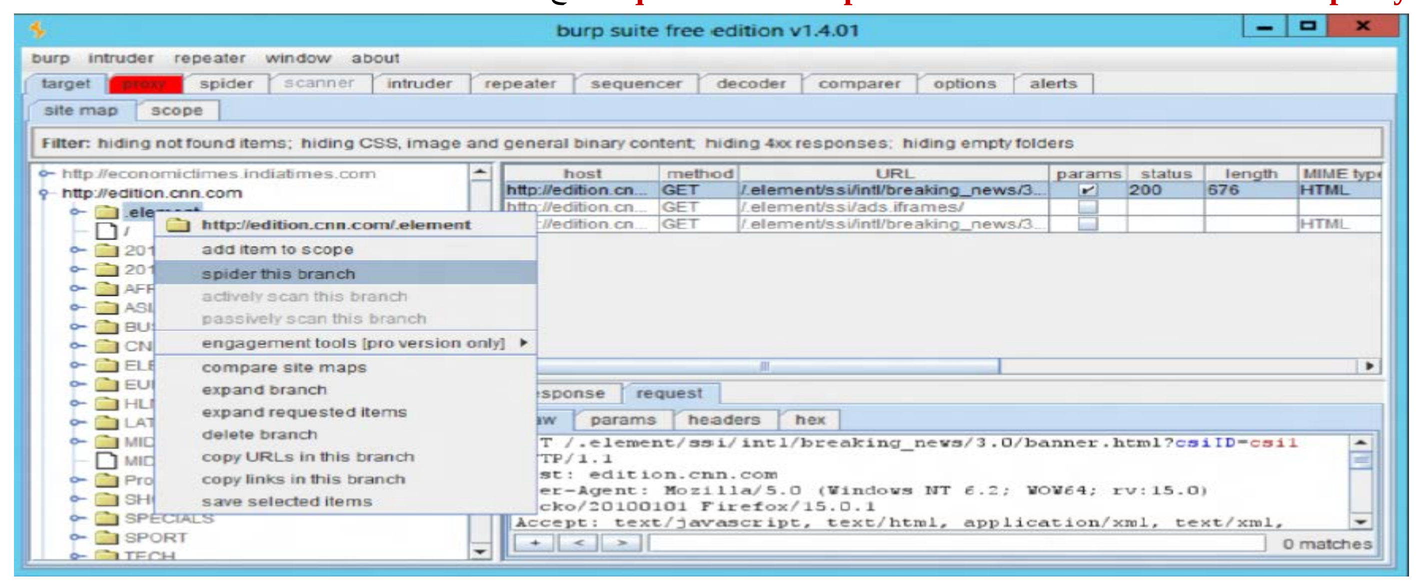
Web Server Attack Methodology: Session Hijacking

اختطاف الجلسة من الممكن ان يتم التعرف على الجلسة الحالية للعميل. السيطرة الكاملة على جلسة عمل المستخدم يمكنه الاستيلاء عليها من قبل المهاجم بمجرد تأسيس المستخدم مصادقة الاتصال مع الخادم. مع مساعدة من أدوات تنبؤ الرقم المتسلسل، فان المهاجمين يقومون بأداء اختطاف الجلسة. المهاجم، بعد تحديد الجلسة المفتوحة، وتوقع رقم التسلسل من الحزمة التالية فانه يرسل حزم البيانات قبل ان يرسل المستخدم الشرعي الاستجابة مع رقم التسلسل الصحيح. وهكذا، فان المهاجم يؤدي اختطاف الجلسة. وبالإضافة إلى هذه التقنية، يمكنك أيضا استخدام غيرها من تقنيات اختطاف الجلسة مثل **session fixation**، **session sidejacking**، **cross-site scripting**، وما إلى ذلك لالتقاط كوكيز الجلسة صحيحة والمعرفات. وهناك أدوات مختلفة ومستخدمة لاختطاف الجلسة وتشمل **Burp Suite**، **Hamster**، **Firesheep**، الخ.

- **Burp Suite**

المصدر: http://portswigger.net

Burp Suite هو منصة متكاملة لأداء اختبار أمن تطبيقات الويب. أدواتها المختلفة تعمل بسلاسة معا لدعم عملية الاختبار بأكملها، من رسم الخرائط وتحليل سطح الهجوم التطبيق، وصولا إلى إيجاد واستغلال الثغرات الأمنية. المكونات الرئيسية **Burp Suite** تشمل **proxy**، **scanner**، **intruder tool**، **repeater tool**، **sequencer tool**، الخ.



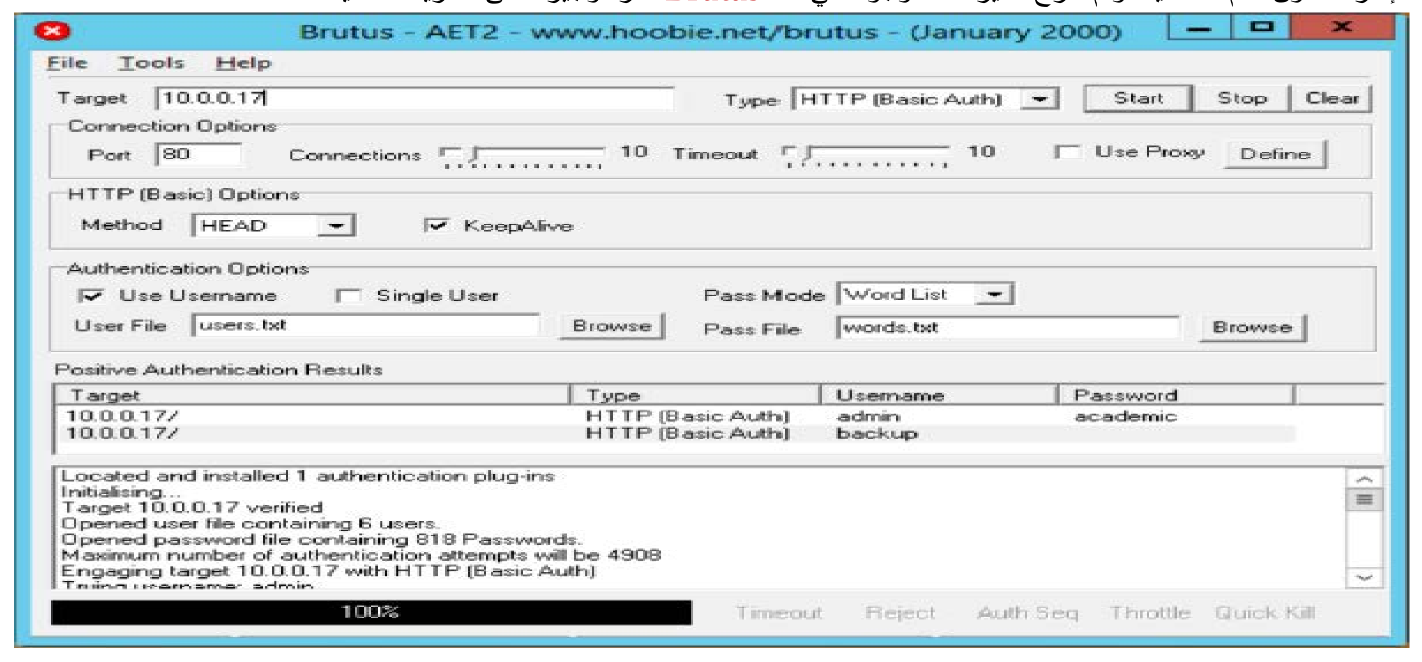
Web Server Attack Methodology: Hacking Web Passwords

واحدة من المهام الرئيسية لأي مهاجم هي قرصنة كلمة السر. عن طريق قرصنة على كلمة السر، فان المهاجم يكسب السيطرة الكاملة على خادم الويب. الطرق المختلفة المستخدمة من قبل المهاجمين لقرصنة كلمة السر تشمل **guessing**، **dictionary attacks**، **brute force attacks**، **hybrid attacks**، **syllable attack**، **precomputed hashes**، **rule-based attacks**، **distributed network attacks**، **rainbow attacks** وما الى ذلك. يمكن أيضا قرصنة كلمة السر مع مساعدة من الأدوات مثل **Brutus**، **THC-Hydra**، الخ.

- **Brutus**

المصدر: http://www.hoobie.net

Brutus هي اداه لكسر كلمات المرور عبر الإنترنت أو عن بعد. المهاجمين يستخدمون هذه الأداة للقرصنة على كلمات السر على شبكة الإنترنت دون علم الضحية. وتم شرح الميزات الموجودة في أداة **Brutus** لفترة وجيزة على الشريحة التالية.



12.4 الأدوات المستخدمة في الهجوم "WEBSERVER ATTACK TOOLS"

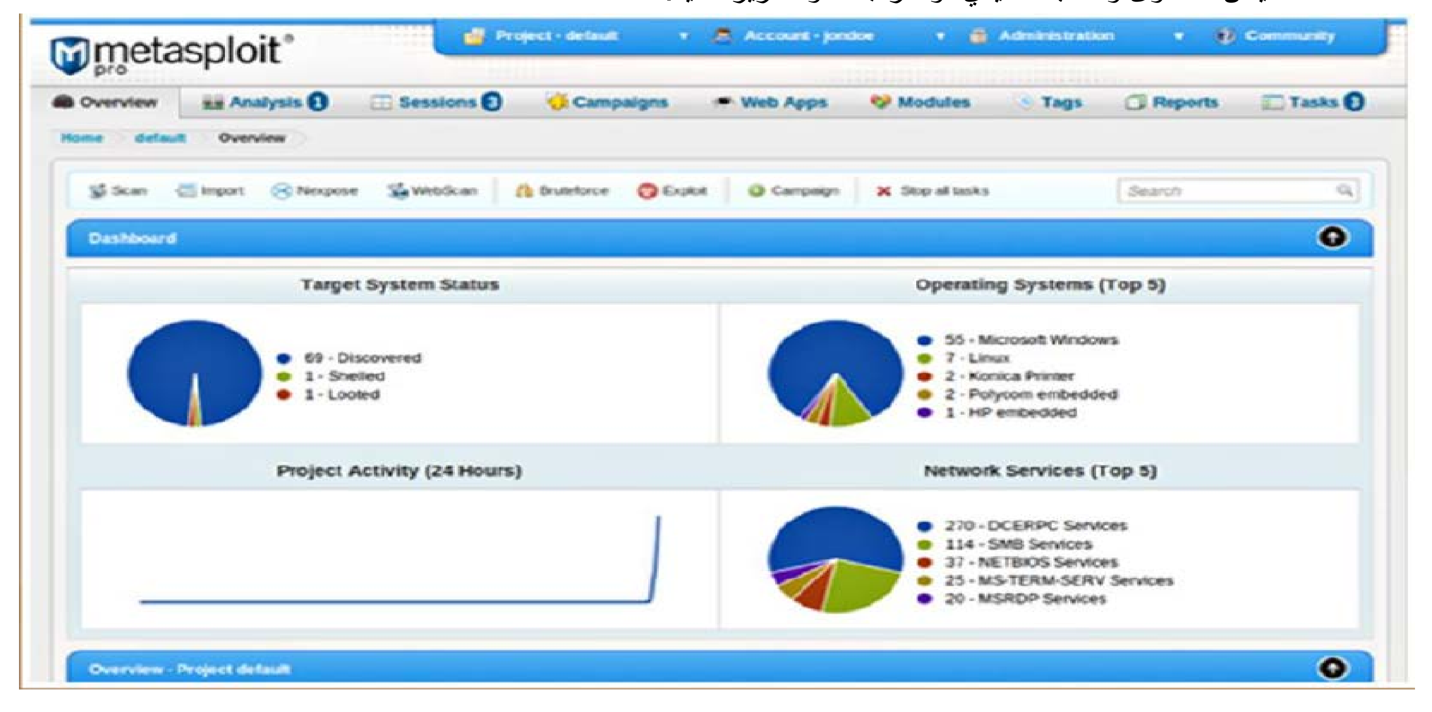
الأدوات المخصصة لمراقبة وإدارة خادم الويب يتم استخدامها أيضا من قبل المهاجمين لأغراض خبيثة. في هذا اليوم وهذا العصر، المهاجمين يقومون بتنفيذ أساليب مختلفة لاختراق خوادم الشبكة. المهاجمين مع الحد الأدنى من المعرفة يستخدمون عادة الأدوات لقرصنة خوادم الشبكة. يسرد هذا القسم ويصف مختلف الأدوات للهجوم على خادم الويب.

WEB SERVER ATTACK TOOLS: METASPLOIT

The Metasploit framework يجعل من اكتشاف، واستغلال، وتقاسم نقاط الضعف سريع وغير مؤلم نسبيا. انها تمكن المستخدمين من تحديد وتقييم واستغلال تطبيقات الويب الضعيفة. باستخدام **VPN pivoting**، يمكنك فاحص نقاط الضعف **NeXpose** من خلال خادم الويب المخترق لاكتشاف نقطة ضعف يمكن استغلالها في قاعدة بيانات الذي يستضيف بيانات العملاء السرية والمعلومات المتعلقة بالموظفين. ويمكن لأعضاء فريقك الاستفادة من البيانات المكتسبة لإجراء الهندسة الاجتماعية في شكل **targeted phishing campaign**، وفتح ناقلات هجوم جديده على الشبكة الداخلية، والتي هي واضحة للفريق بأكمله على الفور. وأخيرا، يمكنك توليد التقارير التنفيذية والتدقيق على أساس قالب الشركات لتمكين مؤسستك للتخفيف من الهجمات وتبقى متوافقة مع **Sarbanes Oxley**، **HIPAA**، أو **PCI DSS**. **Metasploit** تمكن فرق اختبار الاختراق لتنسيق هجمات مدبرة ضد الأنظمة المستهدفة ويؤدي إلى إدارة الوصول المشروع على أساس كل مستخدم. وبالإضافة إلى ذلك، يتضمن **Metasploit** التقرير للتخصيص.

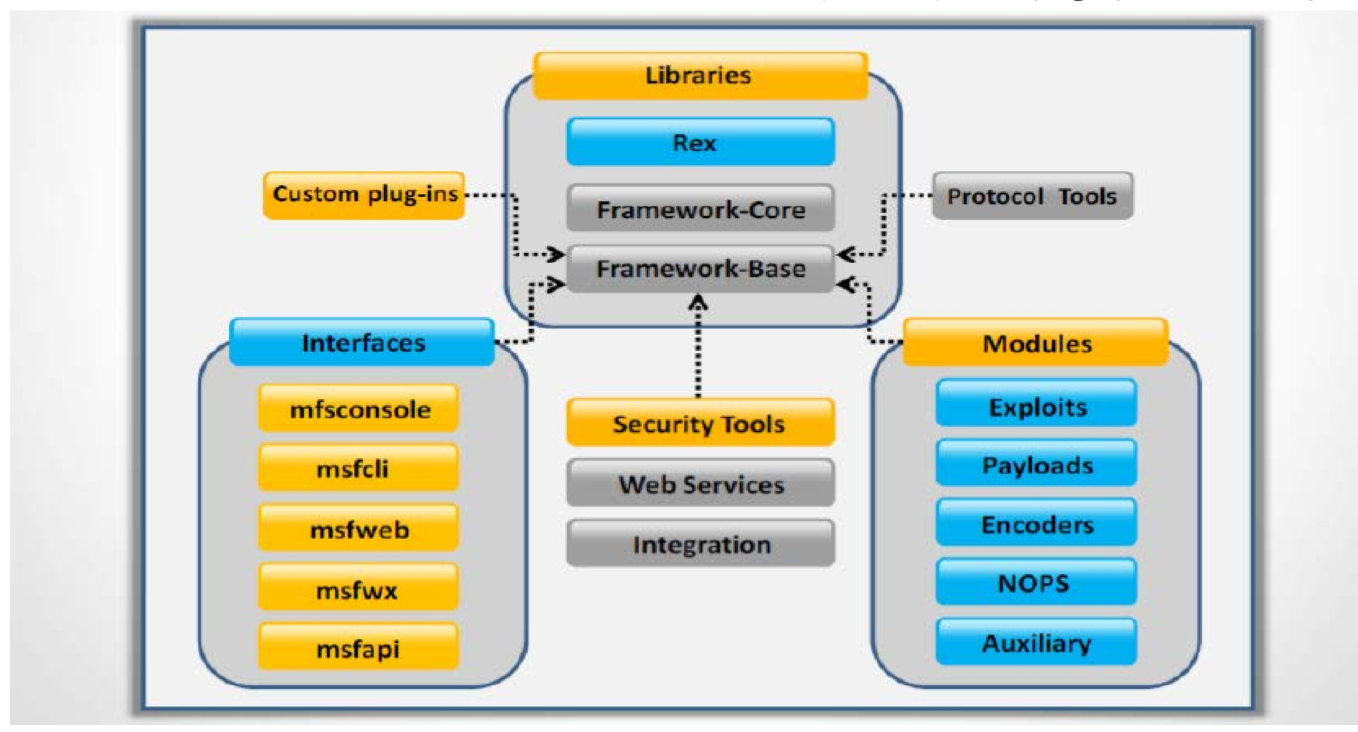
Metasploit تمكنك من:

- اختبار الاختراق كامل بشكل أسرع عن طريق اتمام المهام المتكررة والاستفادة من الهجمات متعدد المستويات.
- تقييم أمن تطبيقات الويب والشبكات وأنظمة نقطة النهاية، وكذلك مستخدمي البريد الإلكتروني.
- محاكاة هجمات الشبكة الواقعية على أساس **Metasploit framework** مع أكثر من مليون من التنزيلات الفريدة من نوعها في العام الماضي.
- اختبار مع أكبر قاعدة بيانات عامة في العالم مضمونة الجودة.
- نفق لأي حركة المرور من خلال الأهداف المخترقة إلى الارتكاز أعمق في الشبكة.
- التعاون بشكل أكثر فعالية مع أعضاء الفريق في اختبارات الشبكة المنسقة.
- تخصيص المحتوى والقالب التنفيذي، والمراجعة، والتقارير الفنية.



معمارية الميتاسبلويت "Metasploit Architecture"

Metasploit framework هو إطار استغلال مفتوح المصدر التي تم تصميمها لتوفير البحوث في مجال الأمن واختبار الاختراق مع نموذج موحد لتحقيق تنمية سريعة لكل من **exploit** و **payload**، **encoders**، مولدات **NOP**، وأدوات الاستطلاع. يوفر هذا الإطار القدرة على إعادة استخدام أجزاء كبيرة من التعليمات البرمجية التي من شأنها أن يكون على خلاف ذلك ليتم نسخها أو **Reimplemented** على الأسس التي تم استغلالها. وقد تم تصميم الإطار لكي يكون وحدات من أجل تشجيع إعادة استخدام الكود عبر مشاريع مختلفة. تم تقسيم الإطار نفسه باستمرار إلى بضع قطع مختلفة، ومعظمها منخفضة المستوى كونها جوهر الإطار. جوهر الإطار هو المسؤول عن تنفيذ كافة الواجهات المطلوبة التي تسمح للتفاعل مع وحدات الاستغلال "**exploit modules**"، **sessions**، و**plugins**. وهو يدعم البحث عن نقاط الضعف واستغلالها للتنمية، وخلق أدوات الأمان المخصص.



Metasploit Exploit Module

وحدة الاستغلال "exploit module" هي وحدة أساسية في **Metasploit** تستخدم لتغليف **exploit** الذي يستخدم في استهداف المستخدمين العديد من المنصات مع **exploit** واحده. هذه الوحدة تأتي مع حقول **meta-information** مبسطة. باستخدام ميزة **Mixins**، يمكن للمستخدمين أيضا تعديل سلوك **exploit** حيويا، تنفيذ هجمات **brute force**، ومحاولة **passive exploits**.

فيما يلي الخطوات لاستغلال النظام باستخدام إطار Metasploit:

- اعداد **active exploit**.
- التحقق من خيارات **exploit**.
- اختيار الهدف.
- اختيار **payload**.
- إطلاق **exploit**.

Metasploit Payload Module

Metasploit payload module تقدم **shellcode** التي يمكنها أن تؤدي عددا من المهام المثيرة للاهتمام للمهاجمين. والحمولة هو قطعة من البرمجيات التي تمكنك من التحكم في نظام الكمبيوتر بعد ان يتم استغلاله. يتم إرفاق الحمولة عادة إلى وتسليمها من قبل **exploit**. **Exploit** يحمل الحمولة في ظهره عند اقتحام النظام ومن ثم يترك ما على ظهره هناك.

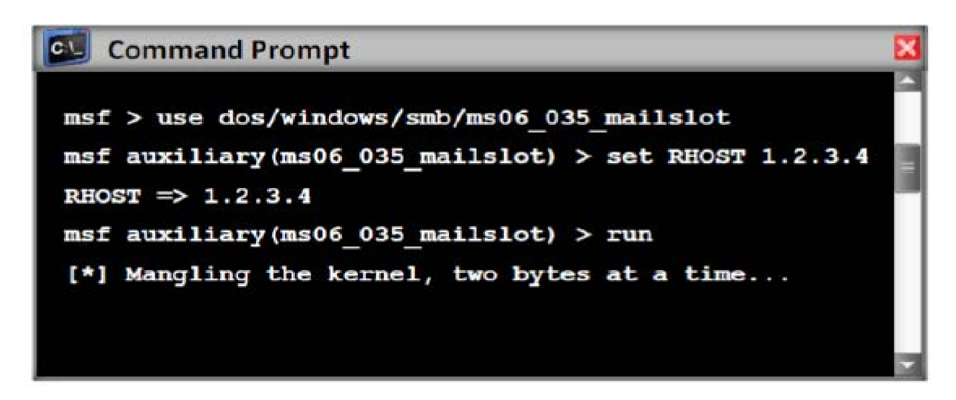
مع مساعدة من الحمولة، يمكنك تحميل وتنزيل الملفات من النظام، واتخاذ لقطات، وجمع **password hashes**. يمكنك حتى التحكم في الشاشة، والماوس، ولوحة المفاتيح للسيطرة الكاملة على جهاز الكمبيوتر.

لتوليد الحمولات "payload"، نحدد أولا الحمولة باستخدام الأوامر:

```
Command Prompt
msf > use windows/shell_reverse_tcp
msf payload(shell_reverse_tcp) > generate -h
Usage: generate [options]
Generates a payload.
OPTIONS:
-b <opt> The list of characters to avoid: '\x00\xff'
-e <opt> The name of the encoder module to use.
-h Help banner.
-o <opt> A comma separated list of options in
         VAR=VAL format.
-s <opt> NOP sled length.
-t <opt> The output type: ruby, perl, c, or raw.
msf payload(shell_reverse_tcp) >
```

Metasploit Auxiliary Module

Metasploit's auxiliary modules يمكن استخدامها لتنفيذ الإجراءات لمرة واحدة والتعسفية مثل فحص المنفذ، والحرمان من الخدمة، وحتى **fuzzing**. لتشغيل **auxiliary module**، إما باستخدام أمر التشغيل أو استخدام أوامر **exploit**.





Metasploit NOPS Module

Metasploit NOP modules تستخدم لتوليد **no operation instructions** التي يمكن استخدامها لحشو **buffers**. واجهة وحدة **NOP** "**NOP module console interface**" يدعم توليد **NOP sled of an arbitrary size** وعرضه في شكل معين.

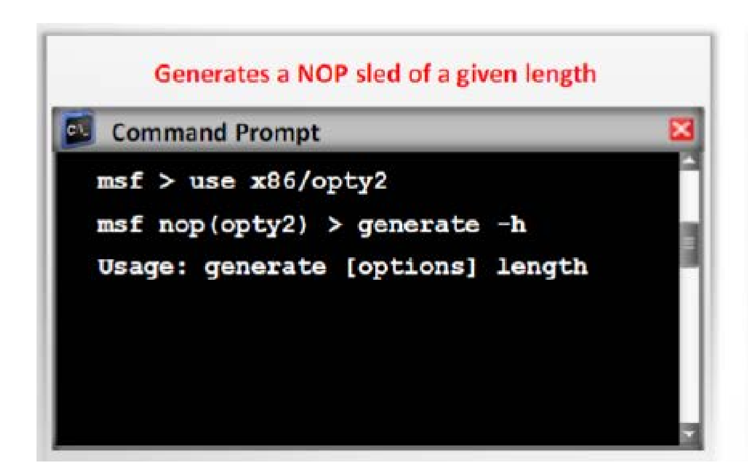
Options:

-b <opt> The list of characters to avoid: ?\xoo\xff?

-h Help banner.

-s <opt> The comma separated list of registers to save.

-t <opt> The output type: ruby, perl, c, or raw.



To generate a 50 byte NOP sled that is displayed as a C-style buffer, run the following command:

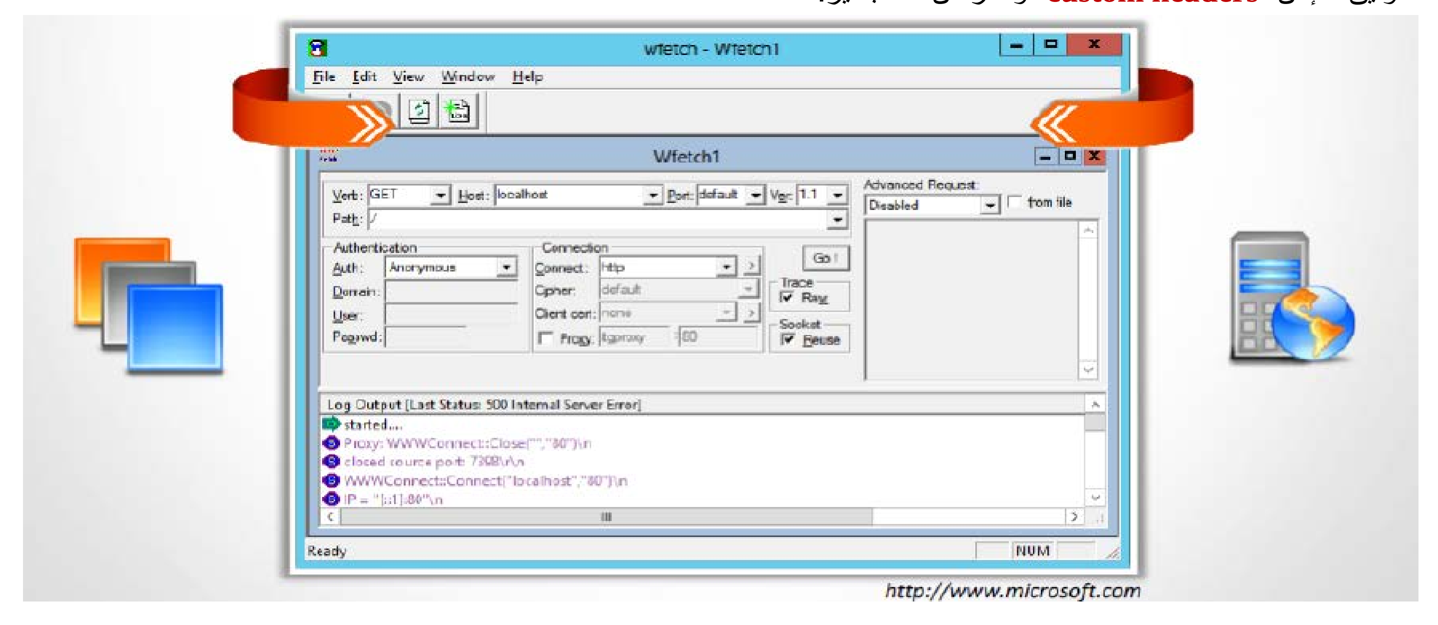
Command Prompt

```
msf nop(opty2) > generate -t c 50
unsigned char buf[] =
"\xf5\x3d\x05\x15\xf8\x67\xba\x7d\x08\xd6\x6
6\x9f\xb8\x2d\xb6"
"\x24\xbe\xb1\x3f\x43\x1d\x93\xb2\x37\x35\x8
4\xd5\x14\x40\xb4"
"\xb3\x41\xb9\x48\x04\x99\x46\xa9\xb0\xb7\x2
f\xfd\x96\x4a\x98"
"\x92\xb5\xd4\x4f\x91";
msf nop(opty2) >
```

WEB SERVER ATTACK TOOLS: Wfetch

المصدر: http://www.microsoft.com/en-eg/default.aspx

Wfetch هي أداة ذات واجه المستخدم الرسومية التي تهدف إلى مساعدة العملاء على حل المشاكل المتعلقة بالتفاعل مع متصفح خادم ويب مايكروسوفت **IIS**. انها تسمح للعميل لإعادة إنتاج مشكلة مع **lightweight**، وبيئة اختبار **HTTP-friendly**. انها تسمح لاختبار التوثيق، الإذن، **custom headers**، وأكثر من ذلك بكثير.



WEB PASSWORD CRACKING TOOL: BRUTUS

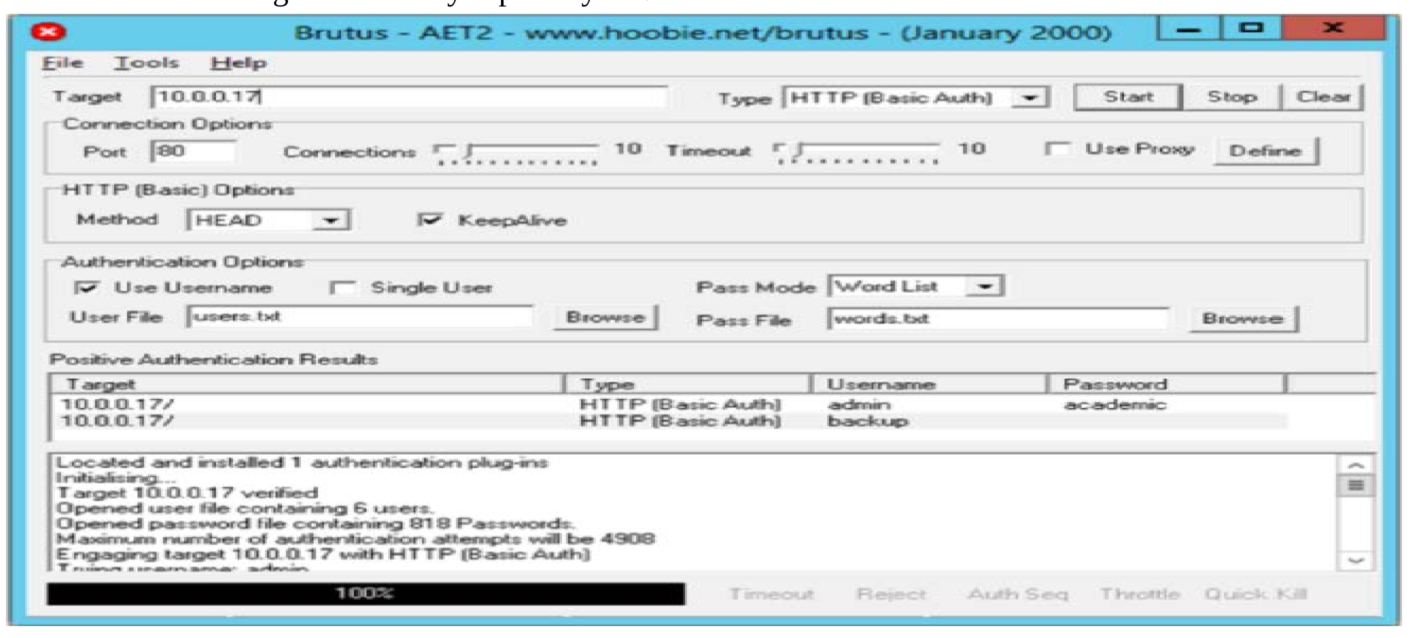
المصدر: http://www.hoobie.net

بروتوس هي أداة لكسر كلمة مرور عن بعد. متوفرا لويندوز **x9**، **NT** و2000، ولا توجد اصدار **UNIX** متاح، على الرغم من أنها محتمله في مرحلة ما في المستقبل. فان بروتوس كتبت في الأصل للمساعدة في التحقق من كلمات مرور الراوتر الافتراضية المشتركة.

المميزات:

- HTTP (Basic Authentication)
- HTTP (HTML Form/CGI)
- POP3
- FTP
- SMB
- Telnet

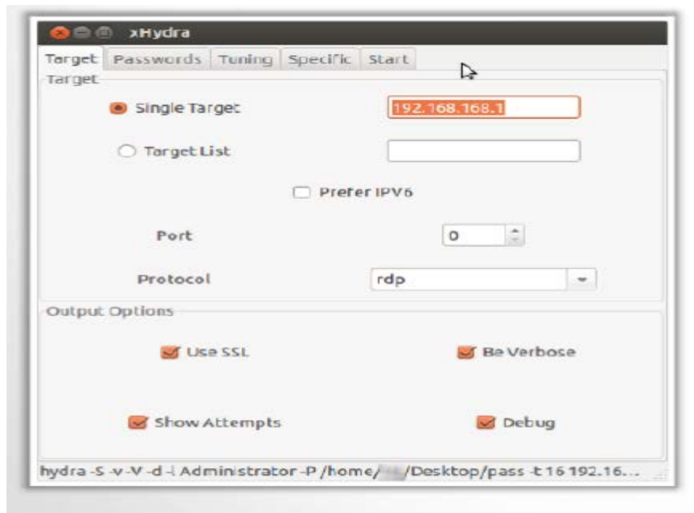
- Multi-stage authentication engine
- No user name, single user name, and multiple user name modes
- Password list, combo (user/password) list and configurable brute force modes
- Highly customizable authentication sequences
- Load and resume position
- Import and Export custom authentication types as BAD files seamlessly
- SOCKS proxy support for all authentication types
- User and password list generation and manipulation functionality
- HTML Form interpretation for HTML Form/CGI authentication types
- Error handling and recovery capability inc. resume after crash/failure

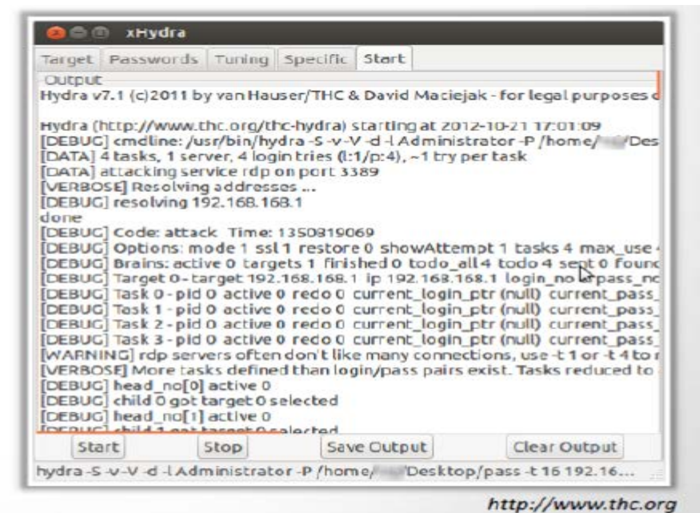


WEB PASSWORD CRACKING TOOL: THC-HYDRA

المصدر: https://www.thc.org

THC-Hydra يستخدم للتحقق من كلمات المرور الضعيفة. هذه الأداة هي أداة **brute force** التي يتم استخدامها من قبل المهاجمين وكذلك الإداريين. **THC-Hydra** يمكنها قرصنة كلمات مرور البريد الإلكتروني والوصول إلى أجهزة الراوتر، أنظمة ويندوز، وخوادم **Telnet** أو **SSH**. وهي لقرصنة كلمات المرور تسجيل الدخول سريعة جدا وتدعم العديد من الخدمات المختلفة.

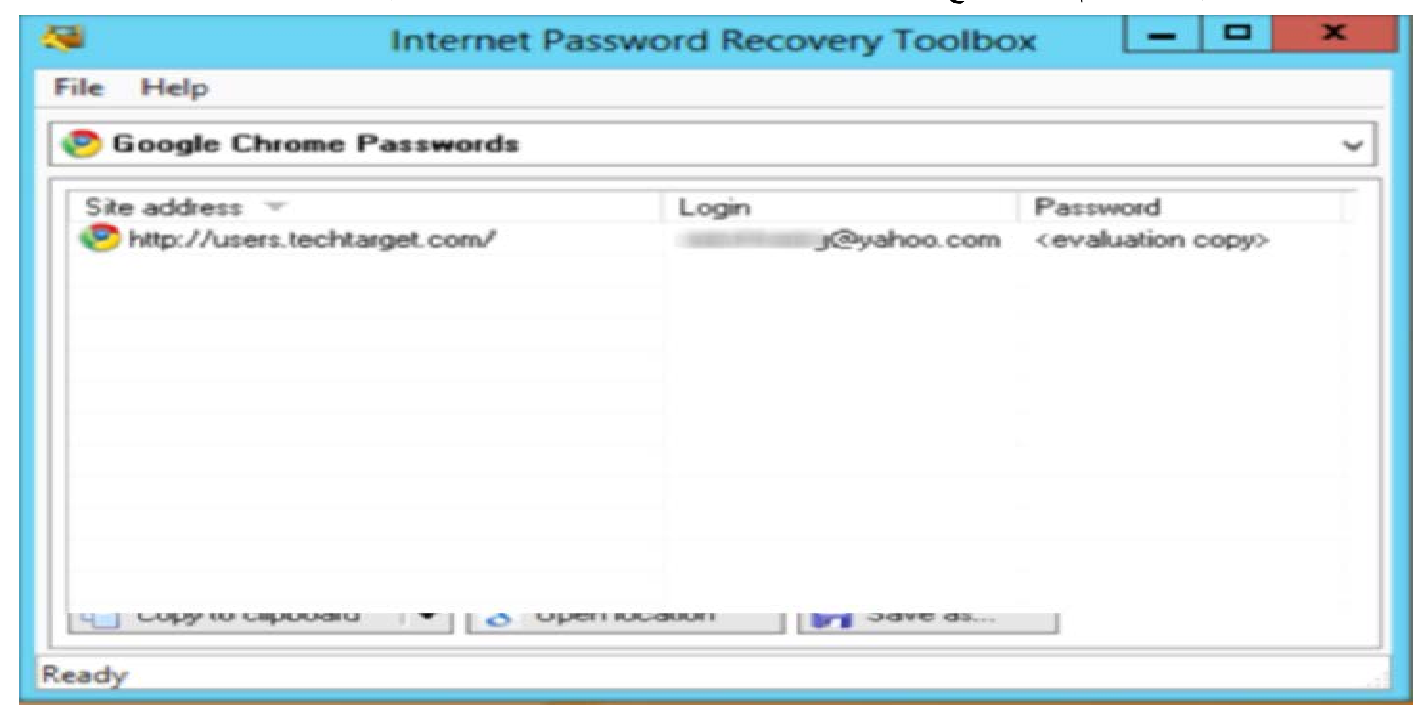




WEB PASSWORD CRACKING TOOL: INTERNET PASSWORD RECOVERY TOOLBOX

المصدر: http://www.rixler.com

Internet Password Recovery Toolbox هو حل شامل لاستعادة كلمات السر لمتصفحات الإنترنت، عملاء البريد الإلكتروني، **nstant messengers**، و **FTP clients**، ويمكنه تغطية الشبكة وحسابات الاتصال الهاتفي، ويمكن استخدامها في المنطقة كلها في وصلات الاتصال بالإنترنت. يقدم هذا البرنامج قدرات استعادة كلمة السر لحظية تقريبا لكل تطبيقات الإنترنت.



12.5 التدابير المضادة "COUNTER-MEASURES"

حتى الآن، قد ناقشنا مفاهيم خادم الويب، والتقنيات المستخدمة من قبل المهاجمين، ومنهجية الهجوم، والأدوات التي تساعد في خادم الويب. كل هذه المفاهيم تساعد في كسر في خادم الويب أو المساس بأمن خادم الويب. الآن حان الوقت لمناقشة التدابير المضادة التي تساعد في تعزيز أمن خوادم الشبكة. التدابير المضادة هي ممارسة استخدام أنظمة أمنية متعددة أو تقنيات لمنع الاقتحام. وهذه هي المكونات الرئيسية للحماية والحفاظ على خادم الويب ضد اختراقات خادم الويب. يسلط هذا القسم الضوء على التدابير المضادة لخادم الويب التي تحمي خوادم الويب ضد الهجمات المختلفة.

COUNTERMEASURES: PATCHES AND UPDATES

وفيما يلي بعض الإجراءات المضادة التي يمكن اعتمادها لحماية خوادم الويب ضد تقنيات القرصنة المختلفة:

- فحص مواطن الضعف القائمة وتصحيح وتحديث برنامج الخادم بشكل منتظم.
- تطبيق كافة التحديثات، بغض النظر عن النوع، "حسب الحاجة".
- ضمان أن حزم الخدمات، الإصلاحات العاجلة، ومستويات تصحيح الأمان تتفق على كافة وحدات تحكم الدومين (DCS). ضمان عند انقطاع ذلك الملقم وجود مجموعة كاملة من الأشرطة الاحتياطية وأقراص الإصلاح الخاصة بحالات الطوارئ متاحة.
- لديك خطة الإعادة التي تسمح للنظام والمشاريع للعودة إلى حالتها الأصلية، وذلك قبل التنفيذ الفاشل.
- قبل تطبيق أي من حزمة الخدمة، الإصلاح، أو تصحيح الأمان، فيجب قراءة ومراجعة نظراء جميع الوثائق ذات الصلة.
- اختبار حزم الخدمات والإصلاحات في بيئة الإنتاج الغير تمثيلية قبل أن يتم نشرهم.
- ضمان اتاحة وجود مجموعة كاملة من الأشرطة الاحتياطية وأقراص الإصلاح لحالات الطوارئ في حالة انقطاع الملقم.
- جدولة ترقيات حزم الخدمة كجزء من عمليات الصيانة.

COUNTERMEASURES: PROTOCOLS

فيما يلي بعض التدابير التي يجب تطبيقها في البروتوكولات ذات الصلة من أجل حماية خوادم الشبكة من القرصنة:

- منع كافة المنافذ الغير ضرورية، حركة مرور **Internet Control Message Protocol (ICMP)**، والبروتوكولات الغير ضرورية مثل **NetBIOS** و**SMB**.
- تصليب "زيادة درجة الامن" **TCP/IP stack** وتفعيل أحدث تصحيحات البرامج باستمرار والتحديثات لبرنامج النظام.
- استخدام بروتوكولات آمنة مثل **Telnet** و**POP3**، **SMTP**، **FTP**، واتخاذ التدابير المناسبة لتوفير مصادقة اتصالات آمنة، على سبيل المثال، استخدام **IPsec policies**.
- إذا كان هناك حاجة الى الوصول البعيد، تأكد من أن يتم تأمين الاتصال البعيد بشكل صحيح، باستخدام نفق وتشفير البروتوكولات
- تعطيل **WEBDAV** إذا لم يستخدم من قبل التطبيق أو الاحتفاظ به آمن إذا كان ذلك مطلوبا.

COUNTERMEASURES: ACCOUNTS

فيما يلي لائحة بالتدابير المضادة ضد قرصنة حسابات خوادم الويب:

- إزالة جميع **modules** الغير مستخدمة و**application extensions**.
- تعطيل حسابات المستخدمين الافتراضي الغير مستخدم والذي تم إنشاؤها أثناء تثبيت نظام التشغيل.
- عند إنشاء **web root directory** جديد على شبكة الإنترنت، منح أذونات **NTFS** (قليله) للمستخدم المجهول ليتم استخدامه من خادم الويب **IIS** للوصول إلى محتوى الويب.
- ازالة قاعدة بيانات المستخدمين الغير ضروري والإجراءات المخزنة واتباع مبدأ الامتيازات الأقل لتطبيق قاعدة الدفاع ضد **SQL query poisoning**.
- استخدام أذونات آمنة على شبكة الإنترنت، أذونات **NTFS**، وآليات مراقبة الدخول لبرنامج **.NET Framework** بما في ذلك **URL authorization**.
- تبطئ هجمات القوة الغاشمة وهجمات القاموس مع سياسات كلمة مرور قوية، ثم التدقيق وحالة التأهب تحسبا لفشل تسجيل الدخول.
- تشغيل العمليات باستخدام على الأقل **privileged accounts** مثل **least privileged service** وحسابات المستخدمين.

COUNTERMEASURES: FILES AND DIRECTORIES

فيما يلي قائمة من الإجراءات التي ينبغي اتخاذها ضد الملفات والمجلدات من أجل حماية خوادم الشبكة من القرصنة:

- القضاء على الملفات غير الضرورية ملفات **within.jar**.
- القضاء على معلومات الاعداد الحساسة ضمن **byte code**.
- تجنب **mapping** المجلدات **virtual** بين اثنين من الخوادم المختلفة أو عبر شبكة الاتصال.
- مراقبة والتحقق من كل سجلات خدمات الشبكة، وسجلات الدخول على الانترنت، وسجلات خادم قاعدة البيانات (على سبيل المثال **Microsoft SQL Server**، **MySQL**، **Oracle**) وسجلات **OS** في كثير من الأحيان.
- تعطيل خدمة قوائم الدليل "**serving of directory listings**".
- ازالة الملفات الغير شبكية مثل ملفات الأرشيف، ملفات النسخ الاحتياطي، الملفات نصية، وملفات **header/include**.
- تعطيل خدمة أنواع معينة من الملفات "**serving certain file types**" من خلال **resource mapping**.
- ضمان وجود تطبيق أو موقع على شبكة الإنترنت والملفات النصية على قسم منفصل أو محرك أقراص غير الموجود عليه نظام التشغيل، والسجلات، وأية من ملفات النظام الأخرى.

كيفية الدفاع ضد الهجمات على خادم ويب

فيما يلي طرق مختلفة للدفاع ضد الهجمات على خادم الويب:

المنافذ "ports"

- التدقيق من المنافذ على الخادم بانتظام لضمان ان الخدمة الغير آمنة أو الغير ضرورية ليست نشطة على خادم الويب الخاص بك.
- الحد من حركة المرور الواردة إلى المنفذ 80 **HTTP** ومنفذ 443 **HTTPS (SSL)**.
- تشفير أو تقييد حركة مرور الشبكة الداخلية.

Server Certificates

- ضمان نطاف بيانات **certificate** صالحة ويتم استخدام **certificate** للغرض المقصود منها.
- ضمان أن **certificate** لا يتم إلغاءها ومفتاح **certificate** العام صالح على طول الطريق إلى سلطة **root** الموثوق بها.

Machine.config

- ضمان أن يتم تعيين الموارد المحمية إلى **HtthorbiddenHandler** ويتم إزالة **HttpModules** الغير مستخدمة.
- التأكد من أن يتم تعطيل التتبع **<trace enable="false"/>** وتعطيل **debug compiles**.

Code Access Security

- تنفيذ الأكواد الأمنه لتجنب الإفصاح عن شفرة المصدر والهجوم على المدخلات.
- تقييد إعدادات السياسة الأمنية للوصول إلى التعليمات البرمجية لضمان ان الأكواد التي يتم تحميلها من الإنترنت لا يوجد لديها أذونات لتنفيذه.
- اعداد **MS** لرفض عناوين المواقع مع "**/..**" لمنع عبور الطريق، وقفل منصة أوامر النظام والمرافق مع تقيد قوائم التحكم بالوصول (قوائم ACL)، وتثبيت التصحيحات الجديدة والتحديثات.

IISLockdown

- **IISLockdown** يقيد الوصول المجهول إلى ادوات النظام، فضلا عن وجود القدرة على الكتابة إلى مجلدات المحتوى على شبكة الإنترنت. للقيام بذلك، **IISLockdown** يخلق مجموعتين محلية جديدة تسمى **web anonymous users** و**web applications**، ومن ثم يقوم بإضافة **deny access control entries (ACES)** لهذه المجموعات إلى قائمة التحكم بالوصول (**ACL**) على المرافق الرئيسية والمجلدات. بعد ذلك، **IISLockdown** يضيف حساب مستخدمي الإنترنت المجهولين الافتراضي (**IUSR_MACHINE**) إلى مستخدمي الويب المجهولين والحساب **IWAM_MACHINE** إلى تطبيقات الويب. إنه يعطل **Web Distributed Authoring** وتعيين الإصدار (**WebDAV**) وتثبيت الفلاتر **ISAPI URLSCAN**.
- استخدام أداة **IISLockdown**، يقلل من ضعف شبكة نظام التشغيل لخادم **Windows 2000**. حيث انها تسمح لك لاختيار نوع معين من دور الخادم، ومن ثم استخدام قوالب مخصصة لتحسين أمن خادم معين.
- **IISLockdown** تثبت الفلتر **ISAPI URLSCAN**، مما يسمح لمسؤولي الموقع لتقييد نوع طلبات **HTTP** التي يمكن للخادم معالجتها، استنادا إلى مجموعة من القواعد فانه يسيطر على الإدارة، ومنع الطلبات الضارة المحتملة من الوصول إلى الخادم والتسبب في الضرر.

Services

- تعطيل الخدمات التي تعمل مع الحسابات الأقل صلاحيات.
- تعطيل **FTP**، **SMTP**، والخدمات **NNTP** إذا لم يكن مطلوبا.
- تعطيل خدمة **telnet**.
- إيقاف كافة الخدمات الغير ضرورية وتعطيلها، لذلك في المرة القادمة التي سوف يتم فيها إعادة تشغيل الملقم، فان هذه الخدمة لا يتم تشغيلها تلقائيا. هذا أيضا يعطي دفعة إضافية لأداء الخادم، من خلال تحرير بعض موارد الأجهزة.

Registry

- تطبيق قوائم التحكم بالوصول المقيدة ومنع إدارة التسجيل عن بعد.
- تأمين **SAM** (الخوادم قائمة بذاتها فقط).

Share

- إزالة جميع مشاركات الملفات الغير ضرورية بما في ذلك مشاركات الإدارة الافتراضية إذا لم يطلب منهم.
- تأمين المشاركة مع أذونات **NTFS** المقيد.

IIS Metabase

- ضمان تكوين تلك الإعدادات المتعلقة بالأمان بشكل مناسب ويتم تقييد الوصول إلى ملف التعريف مع أذونات **NTFS** المؤمنة.
- تقييد معلومات **banner information** التي يتم إرجاعها بواسطة **IIS**.

Auditing and Logging

- تمكين حد أدنى من التدوين على خادم الويب الخاص بك، واستخدام أذونات **NTFS** لحماية ملفات السجل.

Script Mappings

- إزالة كافة **IIS script mappings** الغير ضرورية لامتدادات الملفات الاختيارية لتجنب استغلال أي خلل في ملحقات **ISAPI** التي تتعامل مع هذا النوع من الملفات.

Sites and Virtual Directories

- اعادة مكان المجلدات والمواقع الافتراضية إلى أقسام غير النظام واستخدام أذونات الويب **IIS** لتقييد الوصول.

ISAPI Filters

- حذف فلاتر **ISAPI** الغير ضرورية من خادم الويب.

فيما يلي قائمة من الإجراءات التي يمكن اتخاذها للدفاع عن خوادم الويب من الأنواع المختلفة من الهجمات:

- إنشاء **URL mappings** إلى الخوادم الداخلية بحذر.
- إذا كان خادم قاعدة البيانات مثل **Microsoft SQL Server** مخصص لاستخدامها كقاعدة بيانات الواجهة الخلفية، فيجب تثبيته على ملقم منفصل.
- استخدم آلة مخصصة لخادم الويب.
- لا تثبيت ملقم **IIS** على وحدة تحكم الدومين "**domain controller**".
- استخدام **server-side session ID tracking** ومطابقته الاتصال مع **time stamps**، وعنوان **IP**، الخ.
- استخدام الأدوات الأمنية المتوفرة مع خادم الويب والفاحصات التي تجعل عملية تأمين خادم الويب سهلة.
- مراقبة وفلترة حركة المرور الواردة.
- قم بحماية آلة خادم الويب في غرفة آمنة.
- قم بتكوين حساب مستخدم المجهولين منفصل لكل تطبيق، إذا استضفت تطبيقات ويب متعددة.
- لا تقم بتوصيل ملقم **MS** بالإنترنت حتى يتم تأمينه بالكامل.
- لا تسمح لأي شخص بتسجيل الدخول محليا للدخول إلى الجهاز عدا المسؤول.
- تقييد وظيفة الخادم من أجل دعم تقنيات الويب التي تجري لاستخدامها.

HOW TO DEFEND AGAINST HTTP RESPONSE SPLITTING AND WEB CACHE POISONING

فيما يلي التدابير التي ينبغي اتخاذها من أجل الدفاع ضد HTTP response splitting and web cache poisoning:

Server Admin

- استخدام أحدث برامج خادم الويب.
- التحديث والتصحيح بانتظام لنظام التشغيل وخادم الويب.
- تشغيل فاحص نقاط الضعف للويب.

Application Developers

- تقييد الوصول لتطبيق الويب الى **unique IPS**.
- عدم السماح **carriage return (%0d or \r)** و **line feed (%0a or \n) characters**.
- الامتثال لموصفات **RFC 2616** من اجل **HTTP / 1.1**.

Proxy Servers

- تجنب تشارك اتصالات **TCP** الواردة بين مختلف العملاء.
- استخدام اتصالات **TCP** مختلفة مع البروكسي لمختلف المضيفين الافتراضيين.
- تنفيذ "**maintain request host header**" بشكل صحيح.

12.6 PATCH MANAGEMENT

المطورين دائما يحاولون العثور على **bugs** في خادم الويب ومحاولة اصلاحها. يتم الإفراج عن إصلاحات **bugs** في شكل تصحيحات "**patches**". هذه التصحيحات "**patches**" توفر الحماية ضد نقاط الضعف المعروفة. **Patch management** هو العملية المستخدمة لضمان أن يتم تثبيت **Patch**المناسبة على النظام ومساعدة تحديد نقاط الضعف المعروفة. يصف هذا القسم مفاهيم **Patch management** التي تستخدم لتحديد نقاط الضعف والخلل في خوادم الويب من أجل حمايتهم من الهجمات.

PATCHES AND HOTFIXES

التصحيح هو برنامج يستخدم لإجراء تغييرات في البرامج المثبتة على جهاز الكمبيوتر. وتستخدم التصحيح لإصلاح الخلل، معالجة المشاكل الأمنية، لإضافة وظائف، وما إلى ذلك. التصحيح هو قطعة صغيرة من البرمجيات المصممة لإصلاح المشاكل، الثغرات الأمنية، و **bugs** وتحسين قابليتها للاستخدام أو أداء برامج الكمبيوتر أو البيانات الداعمة لها. ويمكن اعتبار التصحيح بانه إعادة تصحيح وظائف مشكلة البرمجة. **Hotfix** هو حزمة يتم تضمينها في مختلف الملفات المستخدمة خصيصا لمعالجة مختلف المشاكل من البرمجيات. وتستخدم الإصلاحات العاجلة لإصلاح الخلل في المنتج. يتم تحديث المستخدمين حول أحدث الإصلاحات من قبل الموردين من خلال البريد الالكتروني أو أنها يمكن تحميلها من الموقع الرسمي. الإصلاحات العاجلة هي تحديثا لإصلاح قضية محددة من العملاء وليس توزيعها دائما خارج منظمة العملاء. قد يتم إخطار المستخدمين من خلال رسائل البريد الإلكتروني أو من خلال موقع البائع. **Hotfixes** يتم تعبئته في بعض الأحيان على أنه مجموعة من الإصلاحات تسمى **combined hotfix** أو **service pack**.

What Is Patch Management?

وفقا لـ http://searchenterprisedesktop.techtarget.com، فان **patch management** هي مجال إدارة النظم التي تنطوي على اكتساب واختبار وتركيب **patches** متعددة (تغييرات الكود) إلى نظام الكمبيوتر. أنها تنطوي على ما يلي:

- اختيار، التحقق والاختبار وتطبيق التصحيحات.
- تحديث تصحيحات التطبيق السابقة مع التصحيحات الحالية.

- سرد التصحيحات المطبقة سابقا إلى البرنامج الحالي.
- تسجيل **repositories** أو المخازن، لاختيار **patches** بسهوله.
- تعيين ونشر تصحيحات التطبيق.

Detect-1: من المهم جدا دائما الكشف عن تصحيحات الأمان المفقودة من خلال أدوات الكشف المناسبة. إذا كان هناك أي تأخير في عملية الكشف، فان فرص الهجمات الخبيثة تكون مرتفعة جدا.
Assess-2: بمجرد الانتهاء من عملية الكشف فأنه من الأفضل دائما تقييم مختلف القضايا والعوامل المرتبطة بها المتعلقة بها وأفضل تنفيذ لتلك الاستراتيجيات حيث ان هذه القضايا يمكن خفضها أو القضاء عليها بشكل كبير.
Acquire-3: التصحيح المناسب المطلوبة لإصلاح المشكلات لابد من تحميلها.
Test-4: يقترح دائما أولا أن يتم تثبيت التصحيح المطلوب على أن نظام الاختبار بدلا من النظام الرئيسي لأن هذا يوفر فرصة للتحقق من العواقب المختلفة من التحديث.
Deploy-5: الرقع التي يتم نشرها في النظم بـ **utmost=**، لذلك لا يتأثر أي تطبيق للنظام.
Maintain-6: هو دائما مفيد للحصول على إشعارات حول مختلف نقاط الضعف المحتملة كما يتم الإبلاغ عنها.

تحديد المصادر المناسبة للحصول على التحديثات والرقع

من المهم جدا تحديد المصدر المناسب للحصول على التحديثات والرقع. يجب أن تأخذ الرعاية مع الأمور التالية المتعلقة بـ **patch management**.

- **Patch management** التي تناسب البيئة التشغيلية وأهداف العمل.
- ينبغي التخطيط بشكل صحيح.
- البحث عن التحديثات والتصحيحات المناسبة على مواقع منزل التطبيقات أو بائعي أنظمة التشغيل.
- الطريقة الموصى بها لتتبع القضايا ذات الصلة بالترقيع هو التسجيل في المواقع المنزلة لتلقي التنبيهات.

تركيب الرقع/التصحيحات "patch"

يجب عليك البحث عن الرقعة المناسبة وتثبيته من الإنترنت. يمكن تثبيت الرقع بطريقتين:

- **تثبيت يدوي "manual installation"**

في عملية التثبيت اليدوي، المستخدم يقوم بتحميل الرقعة المناسبة من البائع.

- **تركيب التلقائي "automatic installation"**

في التثبيت التلقائي، التطبيقات، مع مساعدة من ميزة التحديث التلقائي، سوف تحصل على التحديث تلقائيا.

التنفيذ والتحقق من امان التصحيح "patch" أو الترقية "update"

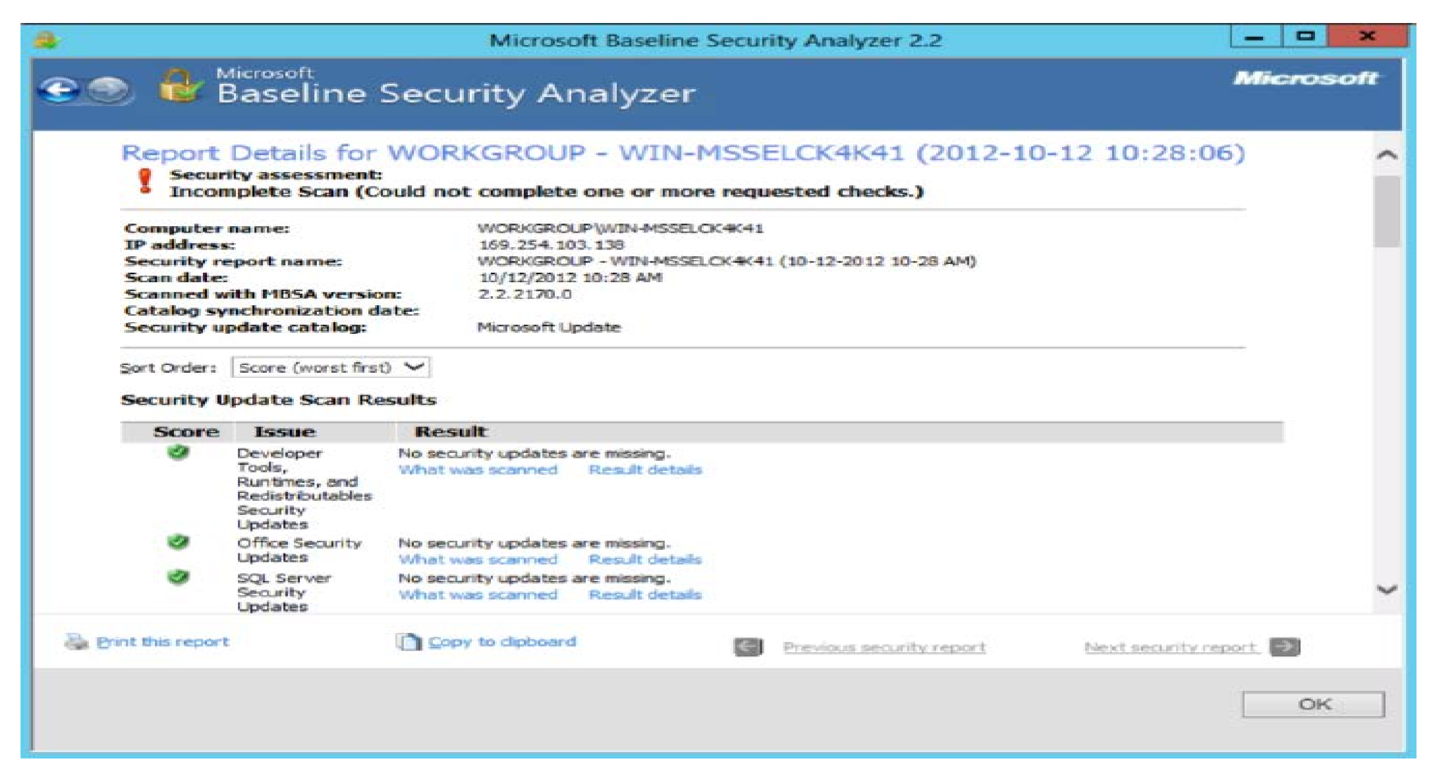
يجب أن تكون على بينة من عدد قليل من الأشياء قبل تنفيذ التصحيح. يجب أن تبقى الأمور التالية في الاعتبار:

- قبل تثبيت أي مصدر للتصحيح، فإنه ينبغي التحقق منه بشكل صحيح. استخدام برنامج إدارة التصحيح السليم للتحقق من صحة إصدارات الملفات واختباره قبل نشر تصحيحات الأمان.
- يجب على فريق إدارة التصحيح التحقق من وجود تحديثات ورقع بانتظام. يجب أن يكون أداة لإدارة التصحيح قادرة على رصد النظم مصححة.

Patch Management Tool: Microsoft Baseline Security Analyzer (MBSA)

المصدر: http://www.microsoft.com

The Microsoft Baseline Security Analyzer (MBSA) يسمح لك لتحديد التحديثات الأمنية المفقودة والاعدادات الخاطئة المشهورة. هو أداة مصممة لمتخصصي تكنولوجيا المعلومات التي تساعد الشركات الصغيرة والمتوسطة الحجم في تحديد حالة أمنهم وفقا للتوصيات الأمنية لمايكروسوفت، ويقدم التوجيه لعلاج محدد. تحسين عملية إدارة الأمن باستخدام **MBSA** للكشف عن الاخطاء الأمنية الشائعة والتحديثات الأمنية المفقودة على أنظمة الكمبيوتر.



- **Patch Management Tools**

بالإضافة إلى **MBSA**، هناك العديد من الأدوات الأخرى التي يمكن استخدامها لتحديد الرقع المفقودة، تحديثات الأمان، واعداد الأمان الخاطئة الشائعة. فيما يلي قائمة بأدوات إدارة التصحيح كالتالي:

Altiris Client Management Suite available at http://www.symantec.com
GFI LANguard available at http://www.gfi.com
Kaseya Security Patch Management available at http://www.kaseya.com
ZENworks Patch Management available at http://www.novell.com
Security Manager Plus available at http://www.manageengine.com
Prism Patch Manager available at http://www.newboundary.com
MaaS360 Patch Analyzer Tool available at http://www.maas360.com
Secunia CSI available at http://secunia.com
Lumension Patch and Remediation available at http://www.lumension.com
VMware vCenter Protect available at http://www.vmware.com

12.7 WEBSERVER SECURITY TOOLS

ينبغي دائما تأمين خوادم الشبكة في بيئة الحوسبة الشبكية لتجنب خطر التعرض للهجوم. أمن خادم الويب يمكن رصدها وإدارتها بمساعدة من أدوات الأمن الخاصة بخادم الويب. يسرد هذا القسم ويصف مختلف الأدوات الأمنية لخادم الويب.

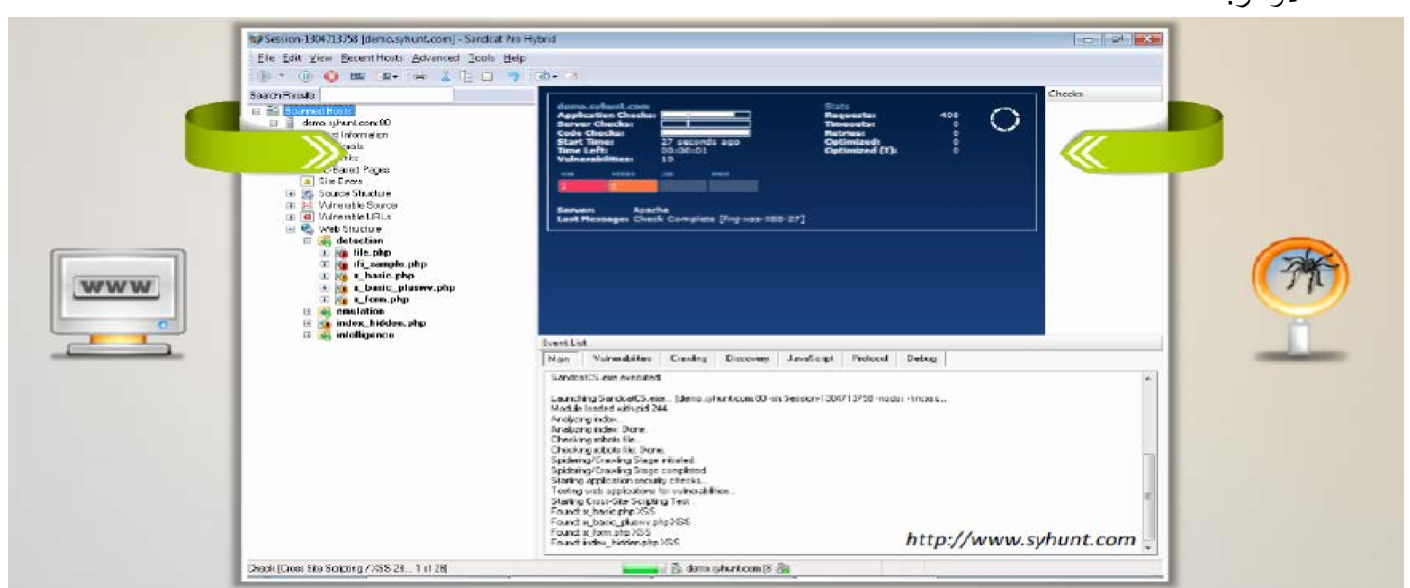
WEB APPLICATION SECURITY SCANNER: SYHUNT DYNAMIC

المصدر: http://www.syhunt.com

Syhunt Dynamic يساعد على الاختبار الأمني لتطبيقات الويب وحراسة البنية التحتية للويب الخاصة بالمنظمة ضد مختلف التهديدات الأمنية لتطبيق الويب.

الميزات:

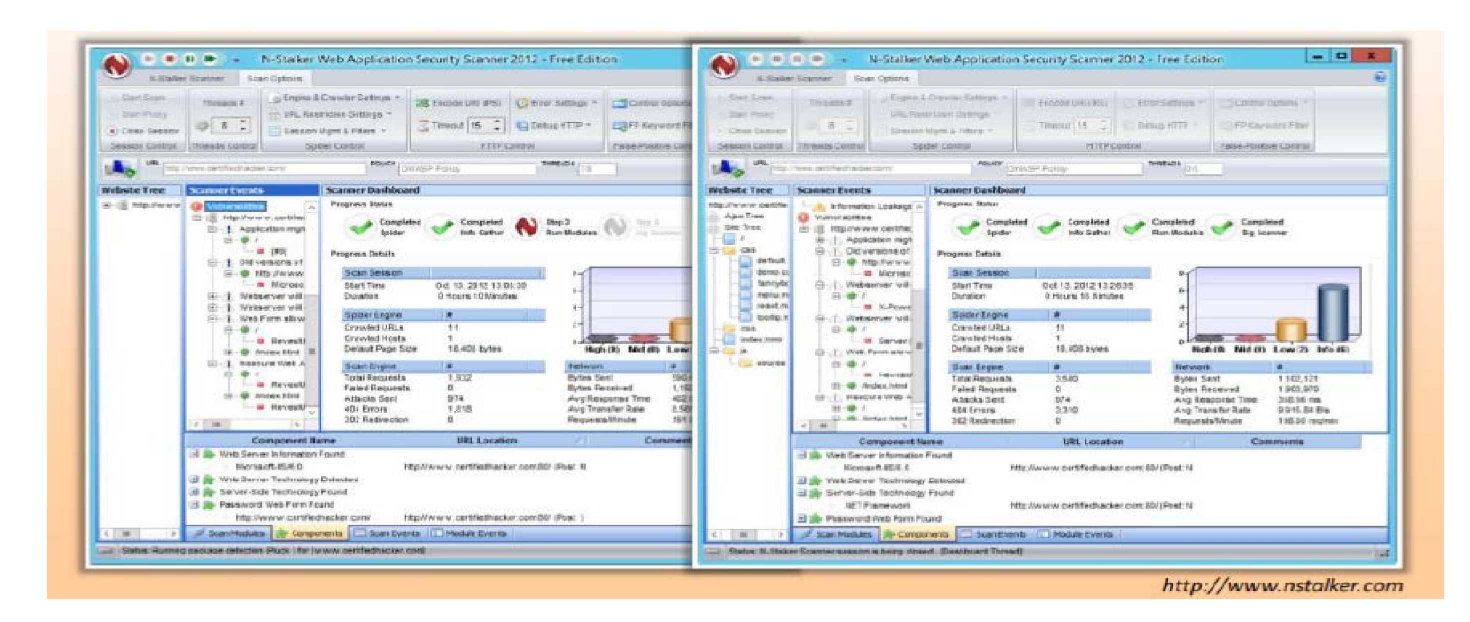
- **Black Box Testing**- تقييم أمن تطبيقات الويب من خلال الفحص عن بعد. وتؤيد أي منصة لخادم الويب.
- **White-Box Testing**- من خلال أتمته عملية مراجعة التعليمات البرمجية للتطبيق على شبكة الإنترنت، فان وظيفة **Sandcat's code scanning** من الممكن جعل حياة المختبر **QA** أسهل، ومساعدتهم على العثور بسرعة والقضاء على الثغرات الأمنية في تطبيقات الويب. يدعم **ASP**، **ASP.NET**، و**PHP**.
- **Concurrency/Scan Queue Support**- الفحص الأمني المتعدد يمكن أن يكون قائمة الانتظار وعدد من المواضيع يمكن تعديلها.
- **Deep Crawling**- يدير الاختبارات الأمنية ضد صفحات الويب التي اكتشفها من خلال الزحف على **URL** واحد أو مجموعة من عناوين المواقع المقدمة من قبل المستخدم.
- **Advanced Injection**- يسرد "**mapping**" هيكل الموقع بأكمله (جميع الروابط والنماذج وطلبات **XHR**، ونقاط الدخول الأخرى) ويحاول أن يجد نقاط الضعف الفريدة من نوعها من خلال محاكاة مجموعة واسعة من الهجمات/إرسال الآلاف من طلبات ومعظمها **GET** و **POST**). اختبارات **SQL Injection**، **XSS**، **File Inclusion**، والعديد من الفئات الأخرى لنقاط الضعف لتطبيق الويب.
- **Reporting** - يولد تقريرا يتضمن معلومات عن نقاط الضعف. بعد فحص استجابة التطبيق للهجمات، إذا تم العثور على **URL** الهدف ضعيفا، فانه يقوم بإضافته إلى التقرير. تتضمن تقارير **Sandcat** أيضا الرسوم البيانية والإحصاءات والمعلومات التوافق. تقدم **Syhunt** مجموعة من قوالب التقرير المصممة لمختلف الجماهير.
- **Local or Remote Storage**- يتم حفظ نتائج الفحص محليا (على القرص) أو عن بعد (في خادم الويب **Sandcat**). ويمكن تحويل النتائج في أي وقت إلى **HTML** أو أشكال أخرى متاحة.
- بالإضافة إلى واجهة المستخدم الرسومية لها (واجهة المستخدم الرسومية)، **Syhunt** تقدم وسيلة سهلة لاستخدام واجهة سطر الأوامر.



WEB APPLICATION SECURITY SCANNER: N-STALKER WEB APPLICATION SECURITY SCANNER

المصدر: http://www.nstalker.com

N-Stalker Web Application Security Scanner هو حل لتقييم الأمن على شبكة الإنترنت لتطبيقات الويب الخاص بك. إنها وسيلة التقييم الأمني الذي يشمل **N-stealth HTTP security scanner**. فإنه يبحث عن نقاط الضعف مثل **SQL injection**، **XSS**، والهجمات المعروفة. كما أنه يساعد في إدارة أمن خادم الويب وتطبيقات الويب. يتم استخدام هذه الأداة الأمنية من قبل المسؤولين والمطورين، ومنظومة/الأمن، ومدققي تكنولوجيا المعلومات، والموظفين.

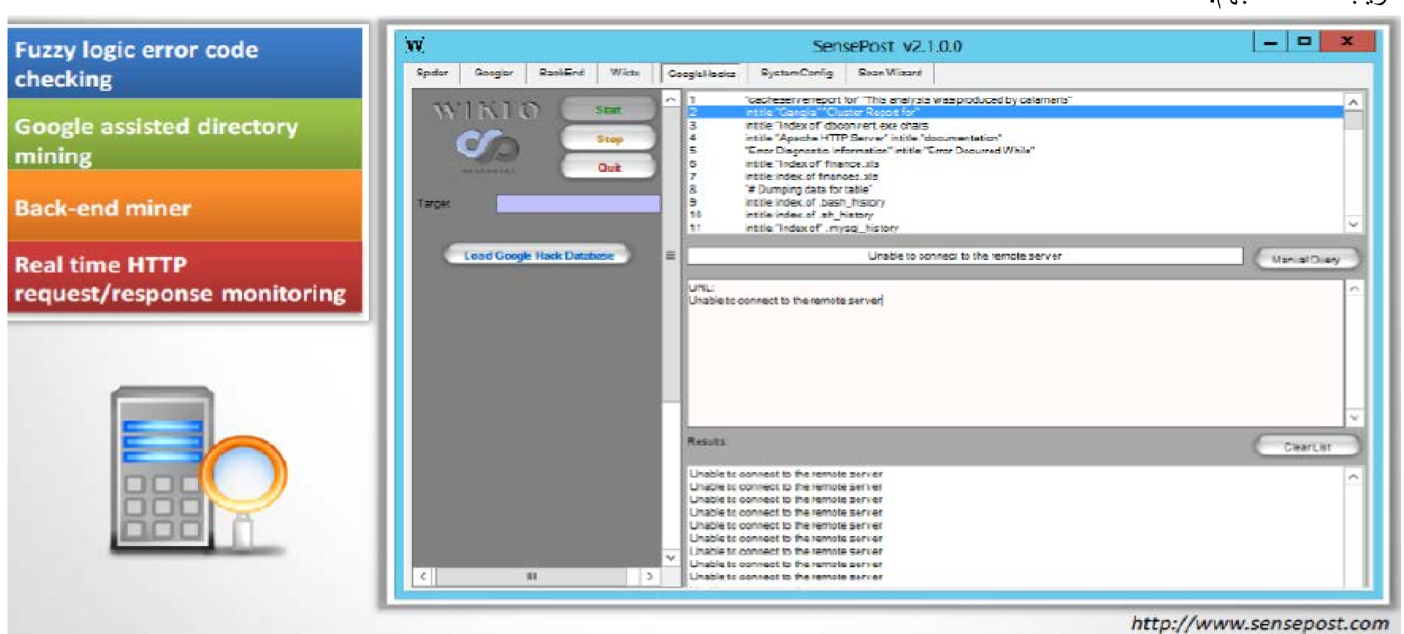


WEB SERVER SECURITY SCANNER: WIKTO

المصدر: http://www.sensepost.com

Wiktoهي اداة ويندوز، مع اثنين من الميزات الاضافية بما في ذلك التحقق من **fuzzy logic error code**، **backend miner**، **Google-assisted directory mining**، ورصد **real-time HTTP request/response**. لقد تم ترميز **Wikto** في **C#**، لذلك يتطلب **.NET framework**.

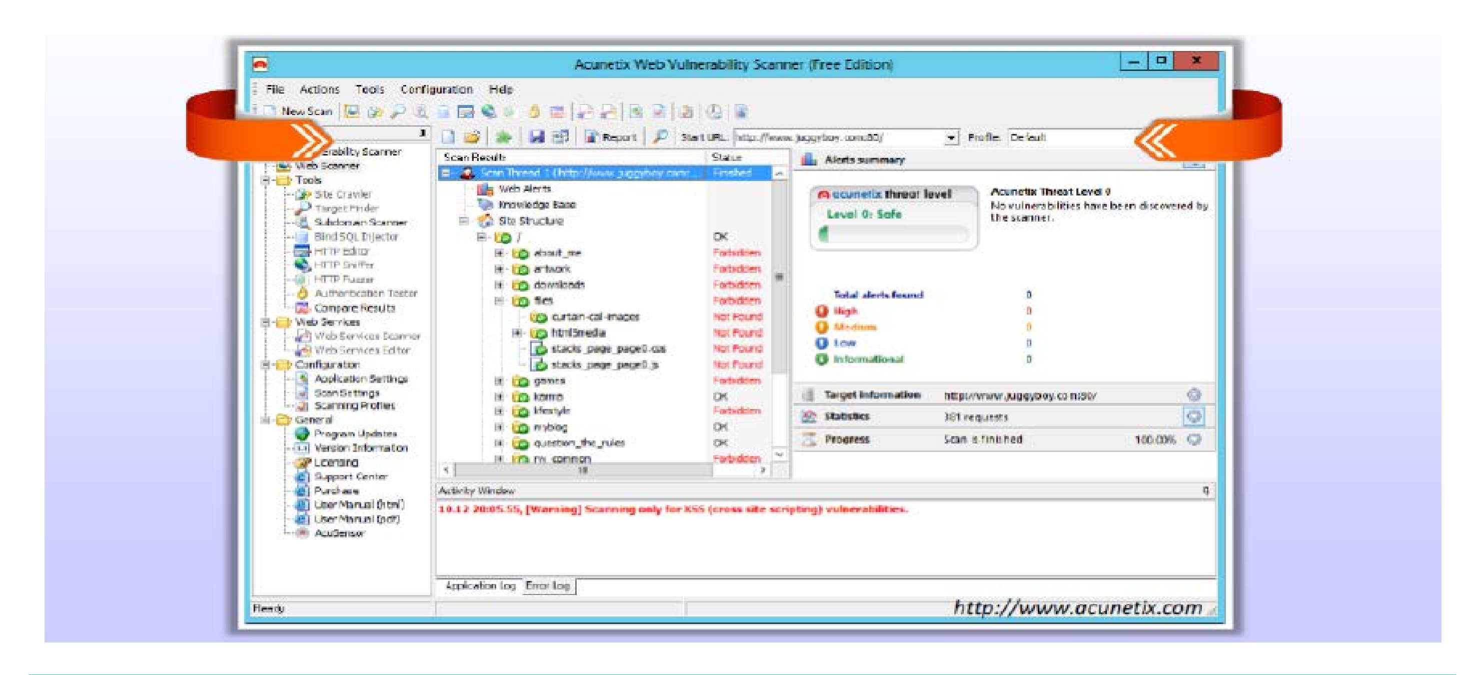
Wikto ربما لا يستخدم لاختبار **SQL injection**، لكنه لا يزال أداة أساسية لاختبار الاختراق الذين يبحثون عن نقاط الضعف في خوادم الويب الخاصة بهم.



WEB SERVER SECURITY SCANNER: ACUNETIX WEB VULNERABILITY SCANNER

المصدر: http://www.acunetix.com

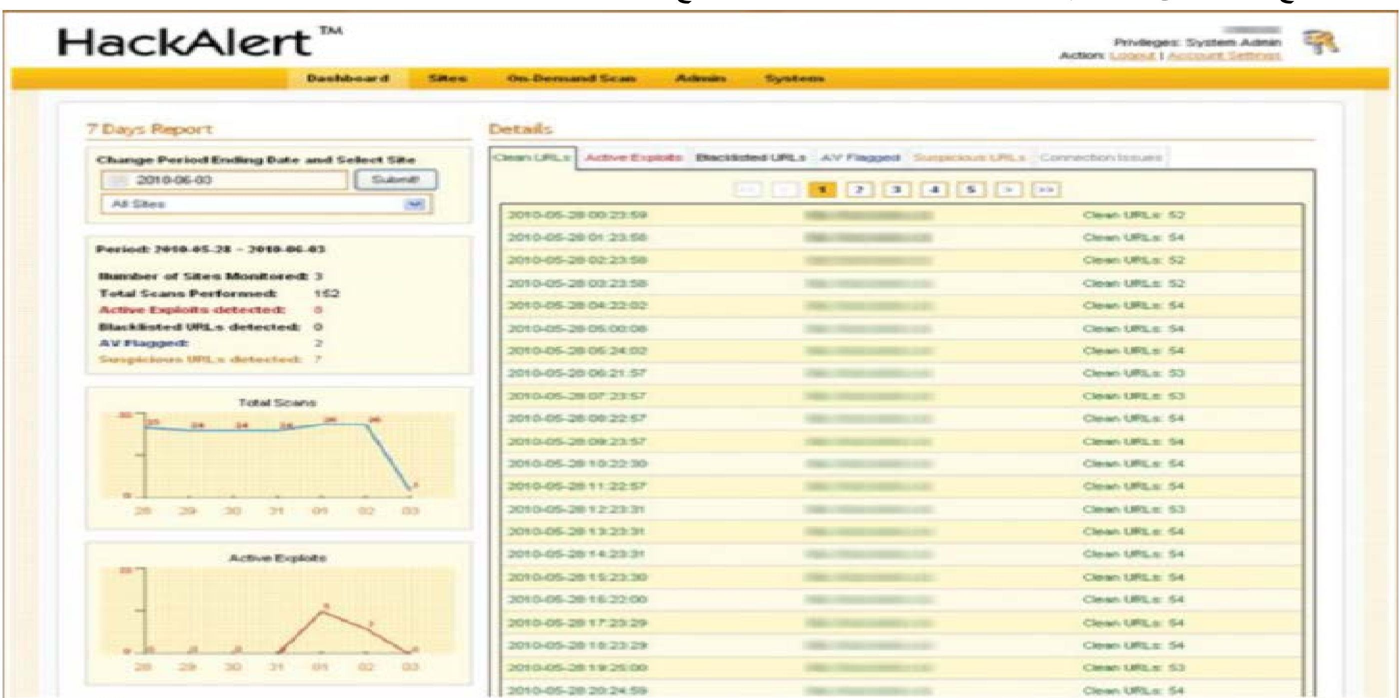
Acunetix Web Vulnerability Scanner يقوم بفحص تطبيقات الويب من **SQL injections**، **cross-site scripting**، وما إلى ذلك. هو يتضمن أدوات متقدمة لاختبار الاختراق للتخفيف من عمليات التدقيق الأمني اليدوية، ويخلق أيضا التدقيق الأمني المحترف والتقارير التنظيمية.



WEB SERVER MALWARE INFECTION MONITORING TOOL: HACKALERT

المصدر: http://www.armorize.com

HackAlert هي خدمة سحابة تحدد **zero-day malware** المخفي و **drive-by downloads** في المواقع والإعلانات عبر الإنترنت. تحسين تقنيات تحليل متعددة، تحدد هذه الخدمة البرمجيات الخبيثة المحقونة ويولد أجهزة الإنذار قبل قيام محركات البحث بسرد الموقع في القائمة السوداء. وهذا يتيح معالجة فورية لحماية العملاء والسمعة التجارية، والإيرادات. يتم الوصول إليه عن طريق إما ادارة العلاقات مع واجهة على شبكة الإنترنت أو **API** مرنة تسهل التكامل مع أدوات المؤسسة الأمنية.

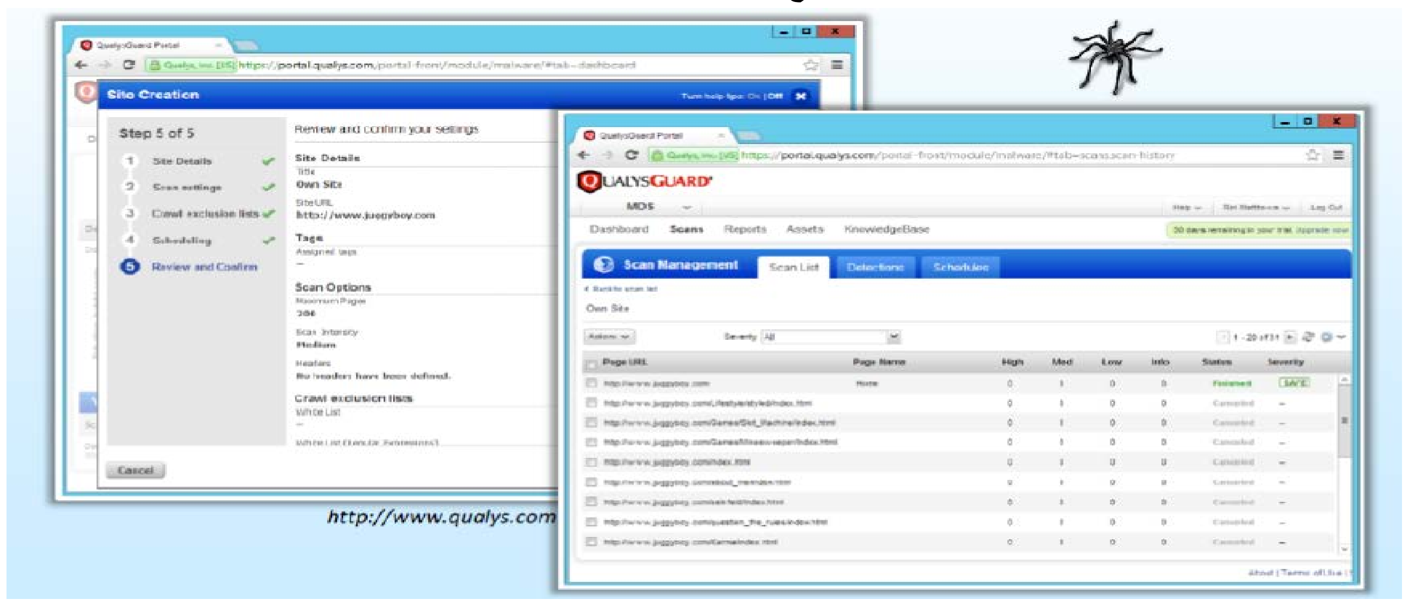


WEB SERVER MALWARE INFECTION MONITORING TOOL: QUALYSGUARD MALWARE DETECTION

المصدر: https://www.qualys.com

QualysGuard Malware Detection هي خدمة فحص المواقع من العدوى الخبيثة ومجموعة متنوعة من التهديدات. ويقدم تنبيهات آليه وتقارير تمكنك من تحديد وإيجاد حل للتهديد. ويمكن أيضا أن تستخدم لحماية عملاء المنظمة من العدوى الخبيثة وحماية سمعة

العلامة التجارية الخاصة، ومنع الموقع من الادراج في القائمة السوداء. بشكل منتظم جداول الفحص لرصد المواقع على أساس مستمر، مع تنبيه عبر البريد الإلكتروني لإعلام المنظمات بسرعة انه تم الكشف عن العدوى. يتم توفير تفاصيل العدوى بالبرمجيات الخبيثة بحيث تمكن المؤسسات من اتخاذ الإجراءات السريعة لعزل وإزالة البرامج الضارة.



WEBSERVER SECURITY TOOLS

أدوات أمن خادم الويب تقوم بفحص، المواقع الكبيرة والمعقدة والتطبيقات على شبكة الإنترنت لمعالجة نقاط الضعف على شبكة الإنترنت. هذه الأدوات تعمل على تحديد نقاط الضعف في التطبيق، فضلا عن خطر التعرض الى الموقع، الترتيب على حسب أولوية التهديد، رسومية عالية، وتقارير **HTML** بديهية، وتشير الى الوضع الأمني للموقع من قبل مواطن الضعف ومستوى التهديد. بعض من أدوات أمن خادم الويب هذه كالاتي:

Retina CS available at http://www.beyondtrust.com
Nscan available at http://nscan.hypermart.net
NetlQ Secure Configuration Manager available at http://www.netiq.com
SAINTScanner available at http://www.saintcorporation.com
HP Weblnspect available at http://download.hpsmartupdate.com
Arirang available at http://monkey.org
N-Stealth Security Scanner available at http://www.nstalker.com
Infiltrator available at http://www.infiltration-systems.com
WebCruiser available at http://sec4app.com
dotDefender available at http://www.applicure.com

WEBSERVER PEN TESTING 12.8

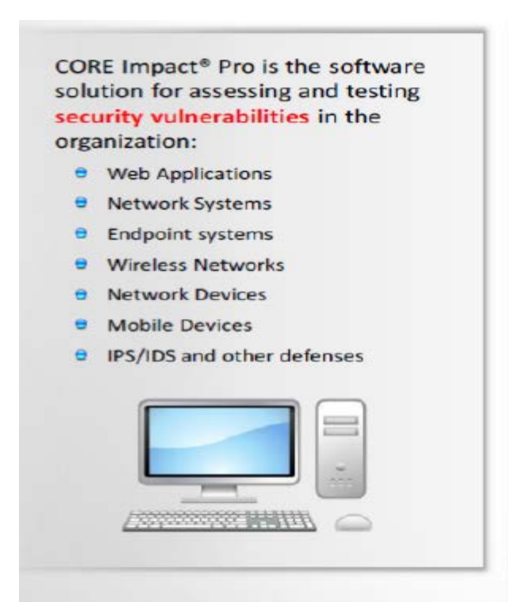
الفكرة كلها وراء الهاكر الأخلاقي هي اختراق الشبكة خاصة بك أو ال في محاولة للعثور على نقاط الضعف ومعالجتها قبل المهاجم الحقيقي. بمثابة إنك مختبر الاختراق، يجب إجراء اختبار الاختراق على خوادم الشبكة من أجل تحديد نقاط الضعف على خادم الويب. يجب تطبيق كل تقنيات القرصنة لقرصنة خوادم الشبكة. يصف هذا القسم أدوات اختبار الاختراق والخطوات المتبعة لاختبار الاختراق لخادم الويب.

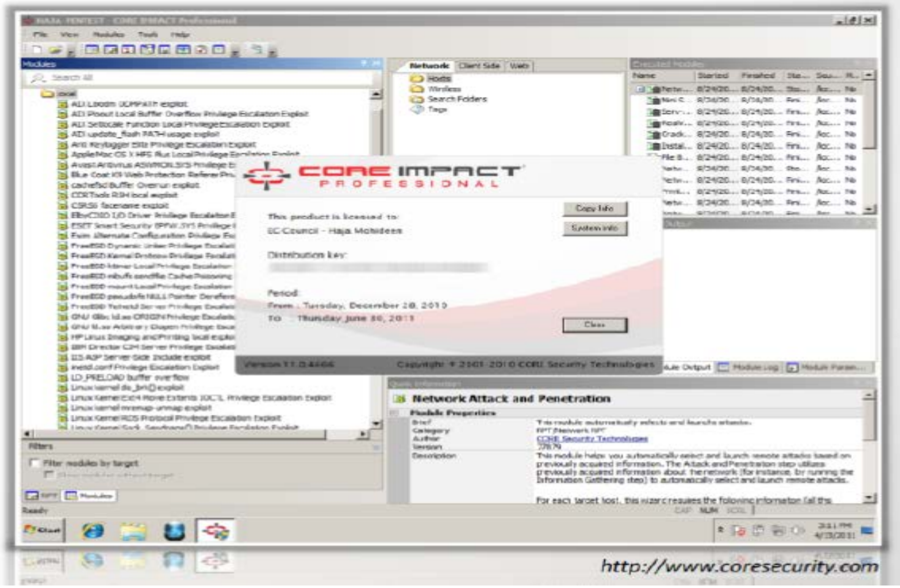
Web Server Pen Testing Tool: CORE Impact® pro

المصدر: http://www.coresecurity.com

CORE Impact® Pro يساعدك في اختراق خوادم الويب للعثور على الثغرات/نقاط الضعف في خادم الويب. من خلال استغلال نقاط الضعف بأمان في البنية التحتية للشبكة الخاصة بك، وتحدد هذه الأداة الحقيقية والمخاطر الملموسة لأصول المعلومات أثناء اختبار فعالية الأمن الموجودة لديك. هذه الأداة هي قادرة على القيام بما يلي:

- تحديد نقاط الضعف في تطبيقات الويب، خوادم الويب، وقواعد البيانات المرتبطة بها.
- توليد حيوي لل **exploit** الذي يمكنه اختراق نقاط الضعف في النظام الأمني لديك.
- إظهار العواقب المحتملة للقرصنة.
- جمع المعلومات اللازمة لمعالجة القضايا الأمنية ومنع حوادث البيانات.

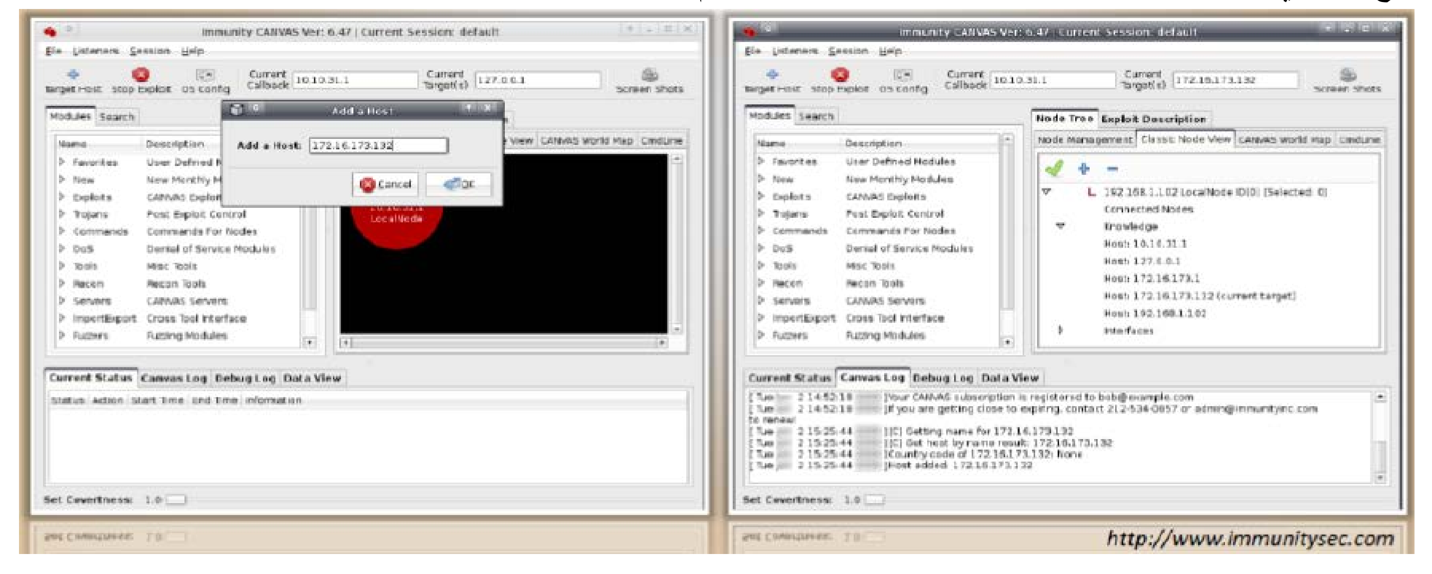




Web Server Pen Testing Tool: Immunity CANVAS

المصدر: http://www.immunitysec.com

CANVAS هو نظام **exploitation** الآلي، وشامل وموثوق به إطار **exploitation** من اجل خبيري الأمن ومختبري الاختراق. لأنها تتيح لمختبري الاختراق اكتشاف كل الثغرات الأمنية المحتملة على خادم الويب.



WEB SERVER PEN TESTING

اختبار الاختراق لخادم الويب تساعدك على تحديد وتحليل وتقديم تقرير عن نقاط الضعف مثل ضعف التوثيق، وأخطاء الاعداد، ونقاط الضعف المتعلقة ب، وما إلى ذلك في خادم الويب. لأداء اختبار الاختراق، تحتاج إلى إجراء سلسلة من الاختبارات المنهجية والمتكررة، والعمل من خلال كل من نقاط ضعف التطبيقات المختلفة.

لماذا يجب على اداء اختبار اختراق خادم الويب

اختبار القلم خادم الويب هو مفيد في الاتي:

- تحديد البنية التحتية للويب: تحديد **make**، نسخة، ومستويات التحديث من خوادم الويب. وهذا يساعد في اختيار **exploit** لاختبار نقاط الضعف المرتبطة المنشورة.
- التحقق من وجود ثغرات أمنية: لاستغلال الضعف من أجل الاختبار وإصلاح المشكلة.
- علاج نقاط الضعف: لإعادة اختبار الحل ضد نقاط الضعف لضمان أنها آمنة تماما.

على شبكة الإنترنت اختبار اختراق الخادم يبدأ مع جمع أكبر قدر من المعلومات الممكنة حول المنظمة، بدءا من موقعه الفعلي لبيئة التشغيل. وفيما يلي سلسلة من الخطوات التي قام بها مختبر الاختبار لاختراق خادم الويب:

الخطوة 1: البحث عن مصادر مفتوحة للحصول على معلومات حول الهدف

في محاولة لجمع أكبر قدر ممكن من المعلومات عن المنظمة الهدف لخادم الويب بدءا من موقعه الفعلي لبيئة التشغيل. يمكنك الحصول على هذه المعلومات من شبكة الإنترنت ومجموعات الأخبار، لوحات الإعلانات، الخ.

الخطوة 2: إجراء الهندسة الاجتماعية

أداء تقنيات الهندسة الاجتماعية لجمع المعلومات مثل الموارد البشرية، وتفاصيل الاتصال، الخ. التي يمكن أن تساعد في اختبار عملية المصادقة لخادم الويب. يمكنك أيضا تنفيذ الهندسة الاجتماعية من خلال مواقع الشبكات الاجتماعية أو القمامة.

الخطوة 3: الاستعلام عن قواعد بيانات Whois

يمكنك استخدام أدوات الاستعلام عن قاعدة بيانات **Whois** مثل **Whois**، **Traceroute**، **Active Whois**، وما إلى ذلك. وذلك للحصول على تفاصيل حول الهدف مثل اسم الدومين وعنوان**IP** ، والاتصالات الإدارية، **Autonomous System Number**، **DNS**، الخ.

الخطوة 4: توثيق جميع المعلومات عن الهدف

يجب توثيق جميع المعلومات التي تم الحصول عليها من المصادر المختلفة.

الخطوة 5: جمع المعلومات عن خادم الويب "Fingerprint the web server"

أداء **Fingerprint** عن خادم الويب لجمع المعلومات مثل اسم الخادم، نوع الخادم وأنظمة التشغيل والتطبيقات قيد التشغيل، الخ باستخدام أدوات مثل **ID Serve**، **httprecon**، **Netcraft**.

الخطوة 6: تنفيذ website crawling

أداء **website crawling** لجمع معلومات محددة من صفحات الويب، مثل عناوين البريد الإلكتروني. يمكنك استخدام أدوات مثل **httprint** و**Metagoofil**.

الخطوة 7: Enumerate web directories

Enumerate web server directories لاستخراج المعلومات الهامة مثل الوظائف على شبكة الإنترنت، **login forms**، وما إلى ذلك. يمكنك القيام بذلك باستخدام أداة مثل **DirBuster**.

الخطوة 8: تنفيذ هجوم directory traversal attack

أداء هجوم **directory traversal attack** للوصول إلى المجلدات المقيدة وتنفيذ الأوامر خارج المجلد الجذري لخادم الإنترنت. يمكنك القيام بذلك عن طريق استخدام الأدوات الآلية مثل **DirBuster**.

الخطوة 9: إجراء الفحص عن نقاط الضعف

أداء الفحص عن نقاط الضعف لتحديد نقاط الضعف في الشبكة باستخدام أدوات مثل **HP WebInspect**، **Nessus**، وغيرها، وتحديد ما إذا كان النظام يمكن استغلاله.

الخطوة 10: تنفيذ هجوم HTTP response splitting

أداء هجوم **HTTP response splitting** لتمرير البيانات الخبيثة للتطبيق ذات نقاط الضعف التي تتضمن البيانات في **HTTP response header**.

الخطوة 11: تنفيذ هجوم web cache poisoning

أداء هجوم **web cache poisoning** على شبكة الإنترنت لإجبار **cache** خادم الويب لطرد محتويات **cache** الفعلي وإرسال طلب وضع خصيصا، والتي سيتم تخزينها في ذاكرة التخزين المؤقت.

الخطوة 12: Brute force login credentials

Brute force SSH، **FTP**، وغيرها من الخدمات وبيانات الدخول للوصول الغير مصرح به.

الخطوة 13: إجراء اختطاف الجلسة "session hijacking"

أداء اختطاف الجلسة لالتقاط كوكيز الجلسة الصحيحة ومعرفات الجلسة. يمكنك استخدام أدوات مثل **Burp Suite**، **Hamster**، **Firesheep**، وما إلى ذلك لأتمتة عملية خطف الجلسة.

الخطوة 14: تنفيذ هجوم رجل في الوسط MITM

أداء هجوم **MITM** للوصول إلى المعلومات الحساسة عن طريق اعتراض وتغيير الاتصالات بين المستخدم النهائي وخوادم الشبكة.

الخطوة 15: تنفيذ اختبار الاختراق لتطبيقات الويب

أداء اختبار الاختراق لتطبيقات الويب لتحديد ما إذا كانت التطبيقات عرضة لنقاط الضعف. تمكن المهاجمين من خرق خادم الويب حتى مع مساعدة من تطبيق ويب ذات نقاط ضعف.

الخطوة 16: فحص سجلات خادم الويب

فحص سجلات الخادم للأنشطة المشبوهة. يمكنك القيام بذلك عن طريق استخدام أدوات مثل **Webalizer**، **AWStats**، **Ktmatu** **Relax**، والخ.

الخطوة 17: Exploit frameworks

Exploit frameworks المستخدمة من قبل خادم الويب باستخدام أدوات مثل **Acunetix**، **Metasploit**، **w3af**، الخ.

الخطوة 18: توثيق جميع النتائج

تلخيص كل التجارب التي أجريت حتى الآن جنبا إلى جنب مع نتائج لمزيد من التحليل. إرسال نسخة من تقرير اختبار الاختراق إلى الشخص المفوض.

الحمد لله تعالى، وبحول الله تعالى نكون قد انتهينا من الوحدة الثانية عشر من CEHv8. ونلقاكم مع الوحدة التالية:

د. محمد صبحي طيبه